



Annika Alakastari

PARIOHJELMOINTI OLIO-OHJELMOINNIN OPETUSMENETELMÄNÄ

PARIOHJELMOINTI OLIO-OHJELMOINNIN OPETUSMENETELMÄNÄ

Annika Alakastari
Opinnäytetyö
Kevät 2013
Tietojenkäsittelyn koulutusohjelma
Oulun seudun ammattikorkeakoulu

TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu
Tietojenkäsittelyn koulutusohjelma

Tekijä(t): Annika Alakastari

Opinnäytetyön nimi: Pariohjelmointi olio-ohjelmoinnin opetusmenetelmänä

Työn ohjaaja(t): Liisa Auer

Työn valmistumislukukausi ja -vuosi: Kevät 2013

Sivumäärä: 43 + 13

Tämän opinnäytetyön tavoitteena oli esitellä pariohjelmoinnin hyötyjä ja haasteita sekä tarkastella sitä opetusmenetelmänä. Pariohjelmoinnilla tarkoitetaan menetelmää, jossa kaksi ohjelmoijaa tekee tiivistä yhteistyötä saman päätteen ääressä toisen tuottaessa lähdekoodia ja toisen korjatessa virheitä ja tehdes-
sä kehitysehdotuksia. Rooleja vaihdetaan säännöllisesti. Varsinaista toimeksiantajaa työllä ei ollut, sillä opinnäytetyön kirjoittaja halusi itse tutkia pariohjelmoinnin hyödyntämistä opetuksessa perehdyttyään me-
netelmään ammattiharjoittelun yhteydessä.

Pariohjelmointia toimivuutta opetusmenetelmänä tutkittiin Tietojenkäsittelyn koulutusohjelman olio-
ohjelmoinnin opintojaksolla. Pariohjelmointikokeilu toteutettiin syksyllä 2012. Opiskelijat jaettiin pareihin
alkukartoituksen perusteella siten, että osapuolten ohjelmointitaidot vastasivat toisiaan. Pareja vaihdettiin
opiskelijoiden halukkuuden mukaan opintojakson puolivälissä. Parit työskentelivät pääsääntöisesti siten,
että rooleja vaihdettiin jokaisen tapaamiskerran aluksi. Opintojaksoon kuuluvat harjoitustehtävät ohjelmoi-
tiin parin kanssa. Harjoitustyöt sekä tentit suoritettiin kuitenkin itsenäisesti. Pariohjelmointi ei ollut pakollis-
ta, mutta sitä suositeltiin kaikille opiskelijoille.

Pariohjelmoinnin käyttämistä opetusmenetelmänä tarkasteltiin niin opiskelijoiden kuin opettajankin näkö-
kulmasta. Opiskelijat vastasivat neljään sähköiseen kyselyyn, joista kolme toteutettiin opintojakson aikana
ja yksi noin kolmen kuukauden kuluttua opintojakson päättymisestä. Kyselyillä kartoitettiin mm. suhtautu-
mista pariohjelmointiin, sen toteutumista ja vaikutusta oppimiseen sekä pariohjelmoinnin hyödyntämistä
opintojakson päättymisen jälkeen. Opettajan havaintoja ja näkemyksiä kartoitettiin haastattelun avulla.
Syksyn 2012 toteutuskertaa verrattiin myös aiempien vuosien toteutuskertoihin oppimistulosten ja läpäisy-
prosentin suhteen.

Tutkimuksen tietoperustana olivat aiemmin tehdyt tutkimukset pariohjelmoinnin hyödyistä ja haasteista
sekä menetelmän käytöstä opetuksessa. Pariohjelmointikokeilun tuloksena havaittiin, että menetelmä so-
veltuu opiskelijoille, joilla on riittävät perusvalmiudet ohjelmoinnissa sekä halua ja kykyä yhteistyöhön.
Opiskelijat, jotka hyödynsivät pariohjelmointia eniten, saivat korkeimmat arvosanat. Vastaavasti opiskelijat,
joilla oli heikko lähtötaso ja vaikeuksia kommunikoida parin kanssa, menestyivät opintojaksolla vaatimat-
tomammin. Suhtautuminen pariohjelmointiin muuttui opintojakson edetessä pääsääntöisesti positiivisem-
maksi, kun opiskelijat havaitsivat hyötyvänsä parin kanssa työskentelystä. Pariohjelmoinnin hyödyntämi-
nen opetuksessa voi antaa opiskelijalle lisävalmiuksia tuleville opintojaksoille sekä työelämään.

Pariohjelmointia voi suositella opetusmenetelmäksi, mutta lisäkokeiluja tarvitaan mahdollisimman toimivan
kokonaisuuden saavuttamiseksi. Parien muodostamisessa voisi kokeilla erilaisia kokoonpanoja. Lisäksi
oppimistuloksia voisi verrata rinnakkain etenevien toteutuskertojen välillä. Toisessa ryhmässä opetus olisi
perinteistä, kun taas toisessa hyödynnettäisiin pariohjelmointia.

Asiasanat: pariohjelmointi, pariohjelmointi opetusmenetelmänä, ohjelmoinnin opettaminen

ABSTRACT

Oulu University of Applied Sciences
Degree programme in Business Information Systems

Author(s): Annika Alakastari

Title of thesis: Pair programming as a Teaching method in Object-oriented programming

Supervisor(s): Liisa Auer

Term and year when the thesis was submitted: Spring 2013

Number of pages: 43 + 13

The purpose of this thesis was to present the costs and benefits of pair programming and to study pair programming as a teaching method. Pair programming is a programming technique in which two programmers use one computer to work together on the same task switching roles as a driver and a co-pilot. There was no actual assigner to this thesis. The author herself wanted to explore pair programming based on her previous experience.

Pair programming as a teaching method was studied in the Degree programme in Business Information Systems in an object-oriented programming course. The pair programming experiment was carried out in the autumn of 2012. The students were divided into pairs on the basis of their programming skills, so that the parties' programming skills mirrored each other. The pairs were changed in the middle of the course. The pairs worked so that the roles were switched in the beginning of each lesson. The course included exercises programmed with a partner. The assignments and exams, however, were performed independently. Programming in pairs was not mandatory, but it was recommended for all students.

The effectiveness of pair programming as a teaching method was examined from both students' and teacher's perspective. Students answered to four electronic surveys, of which three were carried out during the course, and one about three months after the end of the course. The questions asked in the surveys concerned attitude towards pair programming, implementation of pair programming, its impact on learning, as well as using pair programming after the end of the course. The teacher's observations and views were mapped with an interview. The implementation of the course in the autumn 2012 was compared to previous years in terms of learning outcomes and throughput per cent.

The theoretical background of this thesis was based on the earlier studies of pair programming, its cost and benefits, as well as pair programming in education. As a result of this study it was found that pair programming is suitable for students who have sufficient basic skills in programming and willingness and ability to cooperate. Students who used pair programming received the highest marks. On the other hand students with low initial level and difficulties in communicating with a partner, performed modestly. The attitude towards pair programming changed more positive as the course progressed, as the students found to benefit from pair programming. Using pair programming as a teaching method can provide students with additional capabilities for future studies and work.

Pair Programming can be recommended as a teaching method, but further experiments are needed in order to achieve a functional entity. Different combinations could be used in forming pairs. In addition, the learning outcomes could be compared side by side between two course implementations held at the same time. The other group would have traditional teaching, whereas the other would use pair programming as a teaching method.

Keywords: Pair Programming, Pair Programming as a Teaching method, Teaching programming

SISÄLLYS

1 JOHDANTO	6
2 PARIOHJELMOINTI	8
2.1 Pariohjelmoinnin tausta ja määritelmä	8
2.2 Pariohjelmoinnin edut	9
2.3 Pariohjelmoinnin haasteet	10
2.4 Pariohjelmoinnin työskentelyperiaatteet	12
2.5 Pariohjelmointi opetusmenetelmänä	12
3 TUTKITTAVA OPINTOJAKSO	16
3.1 Olio-ohjelmoinnin opintojakson esittely	16
3.2 Pariohjelmoinnin toteuttaminen olio-ohjelmoinnin opintojaksolla	17
4 KYSELYT	18
4.1 Kyselyjen toteuttaminen	18
4.2 Ensimmäinen kysely: Pariohjelmoinnin aloitus	19
4.3 Toinen kysely: Pariohjelmoinnin toteuttaminen	22
4.4 Kolmas kysely: Pariohjelmoinnin päättäminen	25
4.5 Neljäs kysely: Pariohjelmoinnin hyödyntäminen opintojakson päättymisen jälkeen	28
4.6 Opettajan haastattelu	29
5 JOHTOPÄÄTÖKSET	34
6 POHDINTA	38
LÄHTEET	40
LIITTEET	44

1 JOHDANTO

Ohjelmoinnin opiskelu on haastavaa erityisesti silloin, jos aikaisempaa kokemusta ohjelmoinnista ei ole. Ohjelmoinnin opiskelu vaatii loogista ajattelua, keskittymistä, kokonaisuuksien hallitsemista, ongelmanratkaisukykyä, pitkäjänteisyyttä, ohjelmoinnin lainalaisuuksien omaksumista sekä eri ohjelmointikielien syntaksien eli kielioppien omaksumista. Ohjelmointia ei voi opiskella ulkoa tai kopiaimalla, vaan taidot pitää sisäistää. Näistä syistä monet kokevat ohjelmoinnin opiskelun vaativaksi ja työlääksi.

Myös ohjelmoinnin opettamisessa on omat haasteensa. Ohjelmoinnin abstraktien käsitteiden havainnollistaminen opiskelijoille on vaativaa. Lisäksi opettajan pitäisi saada opiskelijat oivaltamaan, että ohjelmoinnin opiskelussa ei ole kyse pelkästään ohjelmakoodin tuottamisesta. Opiskelijoiden pitäisi hallita myös ohjelman tai sovelluksen suunnittelu- ja testausvaiheet, ymmärtää valmista ohjelmakoodia ja oivaltaa dokumentoinnin tärkeys. Usein opiskelijat tyytyvät tuottamaan ohjelmakoodia yrityksen ja erehdyksen kautta, mikä ei auta kokonaisuuden hahmottamisessa ja tuottaa usein uusia virheitä. Opettajan tehtävänä olisi saada opiskelijat ymmärtämään, että ohjelmointi vaatii systemaattisuutta, ennakkointia ja tarkkuutta. Osa ohjelmoinnin opiskelijoista voi olla tottunut ohjelmoimaan pelkästään yksin. Opettajan tehtävänä on muistuttaa opiskelijoita myös siitä, että hyvä ohjelmoija kykenee yhteistyöhön eri osapuolten, kuten asiakkaiden ja kollegoiden kanssa.

Ohjelmointia opiskellaan yleensä opettajajohtoisesti. Opiskelijat työskentelevät itsenäisesti omissa työpisteissään, ja yhteistyö muiden opiskelijoiden kanssa on satunnaista. Opettajalla ei ole aina riittävästi aikaa opiskelijoiden kysymyksiin vastaamiseen, jos ryhmä on suuri. Opiskelijoiden ohjelmointitaidot vaihtelevat erityisesti aikuisopiskelijoiden keskuudessa melkoisesti vasta-alkajista jo ohjelmoinnin parissa työskenteleviin ammattilaisiin. Osalle opintojaksojen vaatimukset ovat todella haasteellisia, kun taas osa turhautuu haasteiden puuttumiseen. Opiskelijoiden vaihtelevat taustat ja ohjelmointitaidot tuovat lisähaastetta opetukseen.

Suorittaessani ammattiharjoittelua Oulun seudun koulutuskuntayhtymän IT-palveluiden (nyk. OSEKK IT) tietohallinnossa pääsin tekemään yhteistyötä toisen harjoittelijan kanssa lähinnä työskentelyolosuhteista johtuen. Vakituksella henkilöstöllä ei ollut välttämättä aina aikaa ja mahdollisuutta harjoittelijoiden opastukseen ja työpisteitä oli niukasti. Näistä syistä päädyimme yhdessä

suunnittelemaan ja toteuttamaan työajanseurantaan liittyvän sovelluksen. Yhteistyö sujui mallikkaasti ja koin oppivani todella paljon jo hieman kokeneemman ohjelmoijan kanssa. Kiinnostuin pariohjelmoinnin tuomista eduista ja ryhdyin perehtymään aiheeseen tarkemmin. Aloin pohtia myös pariohjelmoinnin soveltuvuutta ohjelmoinnin opetukseen johtuen lähinnä omasta opettaja-taustastani. Minua kiinnosti selvittää, auttaisiko pariohjelmointi ohjelmoinnin oppimisessa ja opettamisessa.

Tämän opinnäytetyön tavoitteena on esitellä pariohjelmointia sekä tarkastella sen toimivuutta opetusmenetelmänä Oulun seudun ammattikorkeakoulun (Oamk) Liiketalouden yksikön Tietojenkäsittelyn koulutusohjelman olio-ohjelmoinnin opintojaksolla. Tutkimuksen kohteena ollut opintojakso toteutettiin syksyn 2012 aikana. Soveltuvuutta tarkasteltiin niin opiskelijoiden kuin opettajankin näkökulmasta. Opinnäytetyön tavoitteena on myös antaa ohjelmoinnin opettajille vinkkejä pariohjelmoinnin toteuttamiseen omilla opintojaksoillaan.

Opiskelijoiden suhtautumista pariohjelmointiin ja opintojen edistymistä kartoitettiin neljällä kyselyllä, joista kolme toteutettiin opintojakson aikana ja yksi noin kolme kuukautta opintojakson päättymisen jälkeen. Opintojakson päätyttyä ja arvioinnin valmistuttua opintojakson opettajan haastattelulla selvitettiin myös opettajan havaintoja ja näkemyksiä pariohjelmoinnin soveltuvuudesta kyseiselle opintojaksolle ja ohjelmoinnin opetukseen laajemminkin.

2 PARIOHJELMOINTI

2.1 Pariohjelmoinnin tausta ja määritelmä

Pariohjelmointia on käytetty työskentelymuotona ohjelmistokehityksessä useiden vuosikymmenien ajan. Sen käytänteet vakiintuivat kuitenkin vasta vuoden 2000 tienoilla ketterän ohjelmistokehityksen myötä. Kent Beck ryhtyi kehittämään 1990-luvun lopulla Extreme Programming (XP) – ohjelmistokehitystapaa, jonka keskeiseksi käytännöksi valittiin pariohjelmointi (Beck 1999, 70-77). Williams ja Kessler kirjoittivat vuonna 2003 ensimmäisen pariohjelmointia käsittelevän kirjan, jossa esitellään menetelmän teoreettinen pohja, käytänteet ja ohjenuorat (Williams & Kessler 2003). Vaikka pariohjelmointi on yksi XP:n käytänteistä, se soveltuu käytettäväksi osana mitä tahansa ohjelmistokehitysmenetelmää. Nykyään pariohjelmointia käytetään kyselyiden mukaan useissa ohjelmistotalan yrityksissä (Salo & Abrahamsson 2008, Schindler 2008, Begel & Nagappan 2008).

Pariohjelmoinnissa kaksi ohjelmoijaa tekee tiivistä yhteistyötä. Työskentely tapahtuu yhden tai kahden samaa sisältöä näyttävän päätteen ääressä. Näppäimistöjä on vain yksi. Työpöytä on joko suorakaiteen tai kolmion mallinen. Ohjelmoijat istuvat siten, että molemmilla on esteetön näkymä päätteelle. (Williams & Kessler 2003, 67.) Kuvassa 1 on esimerkki toimistotilasta, jossa kalustus on suunniteltu pariohjelmointia ajatellen.



KUVA 1. Esimerkki pariohjelmointiin soveltuvasta toimistotilasta (Agile Coaching Experience 2013, hakupäivä 30.3.2013)

Toinen ohjelmoijista (driver, controller) on näppäimistön ääressä ja toinen toimii tarkkailijana (co-pilot, navigator, observer). Vuoroja vaihdetaan tasapuolisesti joko tietyin aikavälein tai tilanteen mukaan. Vuoron vaihtuessa vain näppäimistö liikkuu. Driverin tehtävänä on tuottaa ohjelmakoodia. Co-pilot puolestaan korjaa logiikka- ja syntaksivirheitä sekä tekee ohjelmakehitykseen liittyviä ehdotuksia. Pariohjelmointi on jatkuvaa vuorovaikutusta driverin ja co-pilotin välillä. (Beck & Anders 2004, 42.)

2.2 Pariohjelmoinnin edut

Pariohjelmoinnin etuna on mm. ohjelmakoodin virheettömyys. Pareittain tuotettu ohjelmakoodi on myös usein tiiviimpää ja selkeämpää yksin tehtyyn verrattuna. Pariohjelmoinnissa koodia katseloidaan jatkuvasti. Kaksi ohjelmoijaa havaitsee virheet helpommin kuin yksi. Yksin työskennellessä virheet voivat jäädä huomaamatta, koska omalle tuotokselle tulee helposti sokeaksi. Parin kanssa työskennellessä ratkaisut täytyy perustella parille, joten ratkaisuja ja eri vaihtoehtoja tulee mietittyä perusteellisemmin. (Cockburn & Williams 2000, 2-3.)

Pariohjelmointina tuotettujen ohjelmistoprojektien kokonaiskustannukset jäävät useimmiten yksinohjelmoituja pienemmäksi, koska ylläpito- ja tuotetukikustannukset vähenevät virheiden pienen määrän vuoksi. Pariohjelmointina toteutetut projektit ovat siis varsin kustannustehokkaita, vaikka useamman henkilön työpanos nostaakin hieman kustannuksia projektien alkuvaiheessa. (Cockburn & Williams 2000, 4.)

Pariohjelmoinnilla säästetään myös aikaa verrattuna yksin ohjelmoituihin ratkaisuihin, koska pareilta löytyy enemmän ongelmanratkaisukykyä kuin yhdeltä henkilöltä. Voisi olettaa, että tehtävään käytetään kaksinkertainen panos, jos kaksi ihmistä työskentelee yhdessä ratkaisuja pohtien. Tutkimukset osoittavat kuitenkin, että pari käyttää vain noin 15 prosenttia enemmän aikaa ratkaisun tuottamiseen kuin kaksi henkilöä erikseen. (Williams 2001, 27-36). Tutkimuksista on käynyt myös ilmi, että ratkaisujen tuottamiseen tarvittava aika on paremmin ennustettavissa, kun työskennellään pareittain. Tämä auttaa projektien aikataulutuksessa. (Nawrocki & Wojciechowski 2001, 269-276.)

Pariohjelmointi on tehokkainta silloin, kun ohjelmoijat ovat taidoiltaan samalla tasolla, mutta hallitsevat eri osa-alueita. Ohjelmoijat näyttävät toisilleen omat tapansa ratkaista ongelmia ja levittävät omaa osaamistaan jopa koko tiimille, jos pareja vaihdetaan säännöllisesti (Williams & Kessler

2003, 85). Myös ohjelmoijien taidot paranevat parityöskentelyn myötä. Hyvin usein ohjelmoijapari koostuu kuitenkin kokeneemmasta ohjelmoijasta ja vasta-alkajasta, koska pariohjelmointi on osoittautunut tehokkaaksi ohjelmoinnin opiskelumenetelmäksi. Taulukosta 1 käy ilmi erilaisten parien hyötyjä sekä haasteita Williamsin ja Kesslerin (2003, 98-120) mukaan. On kuitenkin huomioitava, ettei taulukosta käy ilmi kaikki mahdolliset yhdistelmät.

TAULUKKO 1. Ohjelmointiparien kokoonpanoja etuineen ja haasteineen (Williams & Kessler 2003, 98-120)

Parin kokoonpano	Etu	Haaste
asiantuntija - asiantuntija	monimutkaiset tehtävät saadaan ratkaistua	vie kaksi tärkeää henkilöresurssia
asiantuntija - aloittelija	aloittelija saa hyvää harjoitusta	asiantuntijan panos kuluu ohjaamiseen
asiantuntija - keskiverto	keskinkertaiset ongelmat saadaan ratkaistua ja keskiverto-ohjelmoijan taidot paranevat	osa asiantuntijan panoksesta kuluu ohjaukseen
aloittelija - aloittelija	aloittelijat saavat arvokasta kokemusta yksinkertaisten tehtävien avulla	aikaa kuluu epäolennaisuuksiin sekä väärin ratkaisuihin

Ohjelmoijan keskittyminen työntekoon on yleensä parempaa silloin kun toinen on mukana seuraamassa työskentelyä. Katkokset työntekoon ja ulkopuoliset häiriötekijät vähenevät, kun tehdään tiivistä yhteistyötä. Yhteistyö varmistaa myös sen, että useampi henkilö hallitsee koodin sisällön. Näin tieto ole vain yhden henkilön varassa, mikä varmistaa sen, että työntekijävaihdoksista huolimatta tieto pysyy myös järjestelmän tuottaneen yrityksen sisällä. (Cockburn & Williams 2000, 8-9.)

2.3 Pariohjelmoinnin haasteet

Yksi pariohjelmoinnin haasteista on tehokkuus. Kaksi ihmistä yhdessä tuottaa vähemmän koodirivejä kuin kaksi ihmistä erillään samassa ajassa. Tosin voidaan pohtia, onko koodirivien määrä ainoa tehokkuuden mitta. Tehokkuus saattaa kärsiä myös silloin, jos pari ajautuu turhaan jaarittelun ja poikkeaa itse ohjelmointityöstä. Vaikka keskittyminen onkin yleensä parempaa toisen kanssa työskennellessä, koska toinen "valvoo" toisen työskentelyä, täytyy molempien osapuolten

olla sitoutuneita tulosten aikaansaamiseksi. (Williams, Kessler, Cunningham & Jeffries 2000, 19-25.)

Pariohjelmoinnissa etätyöskentely on haasteellista, koska parin pitää työskennellä yhdessä saman päätteen ääressä, jotta kaikki pariohjelmoinnin edut saadaan hyödynnettyä. Kaikki ohjelmoijat eivät kuitenkaan pidä ajatuksesta, että joutuvat työskentelemään jatkuvasti jonkun toisen kanssa. Henkilökemioiden pitää kohdata, jotta parin kanssa työskentely tuntuu mielekkäältä. Pariohjelmointi on usein rankkaa, ja siksi monet ohjelmoijat eivät jaksu työskennellä parin kanssa koko työpäivää. Työn tauottaminen ja ylitöiden pitäminen kurissa ovat tärkeitä, jotta pariohjelmointi sujuu. (Sharifabdi & Grot 2002, 166-169.)

Pariohjelmointi soveltuu menetelmänä parhaiten monimutkaisiin ongelmiin. Yksinkertaisemmissa ongelmissa, joihin jo tiedetään ratkaisu, parit kuluttavat enemmän aikaa, ja yksinohjelmointi on näissä tapauksissa tehokkaampaa. Monimutkaisemmassa ohjelmistokehitystyössä, joissa ongelmien ratkaisu ei ole selvillä, useampi henkilö kykenee ratkaisemaan ongelmat tehokkaammin. (Hulkko & Abrahamsson 2005, 495-504.)

Pariohjelmointi vaatii molemmilta ohjelmoijilta panosta tuloksen syntymiseksi. Tarkkailijan rooli on yhtä tärkeä kuin ohjelmakoodin kirjoittajankin. Tarkkailijan keskittyminen ei saa herpaantua toisen koodatessa. Jos parin taidot ovat epäsuhdassa, vaarana voi olla, että vain taidoiltaan parempi ohjelmoija tuottaa koodia ja tekee ratkaisut täysin itsenäisesti ja aloitteleva ohjelmoija jää täysin paitsioon. Tällöin pariohjelmoinnin hyödyt jäävät olemattomiksi. Kommunikointi toisen kanssa on erittäin tärkeää. Pari ei voi työskennellä yhdessä, jos osapuolet eivät kommunikoi keskenään ja jaa ajatuksiaan ohjelmistokehityksen aikana. (Lui & Chan 2006, 915-925.)

Pariohjelmoinnin käyttöönottoa hidastavat usein puutteelliset työskentelytilat. Työpisteet on perinteisesti suunniteltu melko ahtaiksi ja yhtä työntekijää ajatellen. Työskentelytilojen muuttaminen pariohjelmointiin sopivaksi voi olla monelle työnantajalle suuri investointi, koska työpisteiden uudelleenkalustaminen on kallista ja aikaa vievää. Jos työskentelytilat eivät kannusta uuteen työskentelytapaan, jää pariohjelmointi usein pelkäksi kokeiluksi ja muutosvastarinta voi olla korkeaa. Pariohjelmoinnin jalkauttaminen on huomattavasti helpompaa, jos työpisteet on suunniteltu ergonomisiksi kahdelle ohjelmoijalle. (Succi & Marchese 2001, 221-320.)

2.4 Pariohjelmoinnin työskentelyperiaatteet

Jotta pariohjelmointi toimisi työskentelymuotona mahdollisimman hyvin, voidaan työskentelyperiaatteet koota Williamsia ja Kessleriä (2003) mukaillen seuraavasti:

1. Suhtaudu pariisi ja pariohjelmointiin avoimesti. Heitä pois turha skeptisyys.
2. Jaa osaamistasi avoimesti. Samalla opit myös itse uutta.
3. Pelaa reilua peliä. Driverin ja co-pilotin roolit ovat yhtä tärkeitä.
4. Keskity olennaiseen. Tulosta syntyy ainoastaan keskittymällä kehitettävään järjestelmään.
5. Älä aliarvioi ohjelmointikykyäsi. Pariohjelmointi on mainio keino taitojen kehittämiseen.
6. Jokainen tekee virheitä. Co-pilotin tehtävänä ei ole kritisoida driverin tuotosta, vaan edistää kehitystyötä.
7. Älä aliarvioi parisi ohjelmointitaitoja. Kukaan ei ole täydellinen.
8. Muista perustella näkemyksesi. Voit olla eri mieltä parisi kanssa, mutta on tärkeää esittää perustelut.
9. Huolehdi työpisteen soveltuvuudesta parityöskentelyyn. Näytön pitää olla tarpeeksi suuri, ja työtuoleja ei pitäisi tarvita siirtää – vain näppäimistö siirtyy.
10. Pidä parisi kanssa riittävästi taukoja. Pariohjelmointi on hyvin intensiivinen työskentelymuoto.
11. Jätä tilaa myös vapaamuotoiselle ideoinnille. Parin ei tarvitse työskennellä näytön ääressä koko työaika.
12. Työskentele välillä myös itsenäisesti.
13. Molempien osapuolten pitää kantaa vastuu, mutta myös nauttia työn tuloksista.
14. $1 + 1$ on enemmän kuin kaksi. Pari tuottaa enemmän ideoita kuin kaksi henkilöä erillään.

2.5 Pariohjelmointi opetusmenetelmänä

Jokainen oppija on erilainen. Oppiminen on prosessi, johon vaikuttavat esimerkiksi opiskelijan motivaatio, tietopohja, oppimistyyli, oppimisympäristö ja opetustyyli. Myös vuorovaikutuksen merkitys oppimisessa on olennainen. Keskustelemalla ja pohtimalla yhdessä saavutetaan monipuolisempi ja syvällisempi käsitys opiskeltavasta aiheesta. Ihanteellisessa oppimisympäristössä oppija saa laajentaa tietämystään omista lähtökohdistaan vuorovaikutuksessa toisten opiskelijoiden ja

opettajan kanssa. Siten myös pariohjelmointi voi auttaa ohjelmoinnin lainalaisuuksien omaksumisessa.

Korkeakouluissa suoritetuissa pariohjelmointitutkimuksissa on vertailtu parin kanssa ja yksin ohjelmoineiden opintomenestystä ja kokemuksia samoilla opintojaksoilla. Pariohjelmointikokeiluja on toteutettu käyttämällä erilaisia menetelmiä parien muodostukseen. Erilaisten parien työskenteilyä seuraamalla on saatu viitteitä toimivien parien kokoonpanosta. Joissakin kokeiluissa parit ovat opettajien määräämiä. Toisissa tutkimuksissa opiskelijat ovat saaneet itse valita parinsa tai pareja on vaihdettu jatkuvasti. Parien osapuolina on ollut erilaisia persoonallisuuksia, ohjelmointitaitoiltaan erilaisia opiskelijoita sekä eri vuosikurssien opiskelijoita.

Parien muodostuksesta ja vaihdosta on saatu hieman ristiriitaisia tutkimustuloksia. Toisaalta erilaiset temperamentit lisäävät tuottavuutta (Sfetsos, Stamelos, Angelis & Deligiannis 2006, 43-52.), mutta parin erilainen persoona on koettu myös pariohjelmoinnin epäkohdaksi (VanDeGrift 2004, 2-6). Erityisesti taitavien opiskelijoiden kohdalla parin ohjelmointitaitojen vastaavuus vaikuttaa pariohjelmoinnin mielekkyyteen (Thomas, Ratcliffe & Robertsson 2003, 363-367). Tasavertaiset osapuolet ratkaisevat ongelmia tehokkaammin, mutta parilta oppiminen jää vähäisemmäksi (Van Toll III, Lee & Ahlswede 2007, 302-308). Jos opiskelijat saavat itse valita parit, he valitsevat mieluiten samantasoisen parin (Katira, Williams, Wiebe, Miller, Balik & Gehringer 2004, 7-11). Paras tulos saavutetaan, jos opiskelijat saavat valita itse parinsa, jolla on sama motivaatiotaso (Katira, Williams & Osborne 2005, 625-626). Toisaalta työelämässä täytyy pystyä työskentelemään eritasoisten ja erilaisten kollegoiden kanssa (Van Toll III ym. 2007, 302-308). Parit, joissa toinen on mies ja toinen nainen toimivat tutkimustulosten mukaan heikoiten (Katira ym. 2004, 7-11). Parien vaihto koettiin mielekkääksi, koska tällöin oppii tuntemaan useampia opiskelutovereita (Cliburn 2003, 20-29). Paria ei saa vaihtaa kuitenkaan liian usein, koska silloin ei jää riittävästi aikaa työskentelyrytmin löytämiseen (Keefe, Sheard & Dick 2006, 91-100).

Tutkimusten mukaan pariohjelmoijien ja yksin työskennelleiden tenttituloksissa ei ole merkittäviä eroja (Nagappan, Williams, Ferzli, Wiebe, Yang, Miller & Balik 2003, 359-362). Tenttiin osallistuneiden määrissä on kuitenkin havaittu, että parin kanssa ohjelmoineet osallistuivat merkittävästi useammin opintojakson tenttiin kuin yksin työskennelleet (McDowell, Werner, Bullock & Fernald 2006, 90-95). Parin kanssa työskennelleet saivat myös suoritettua opintojakson huomattavasti todennäköisemmin kuin yksin ohjelmoineet (Mendes, Al-Fakhri & Luxton-Reilly 2005, 296-300). Pariohjelmoijat olivat tyytyväisempiä harjoitustöihinsä ja heidän mielestään pariohjelmointi tuotti

laadukkaampaa ohjelmakoodia ja toimivampia ratkaisuja. Harjoitustöistä saadut pisteet tukevat opiskelijoiden näkemystä. Opiskelijat ymmärsivät omasta mielestään tuottamansa ratkaisun ja jäljittivät virheitä paremmin. He myös pitivät harjoitustyön tekemisestä enemmän kuin yksin työskennelleet ja palauttivat harjoitustyöt määräaikaan mennessä. (Williams & Kessler 2000, 59-65.) Parin kanssa ohjelmoineet osallistuivat ja menestyivät pariohjelmointikokeilua seuraavilla ohjelmointikursseilla yksin työskennelleitä paremmin. Erityisesti parin kanssa ohjelmoineet miehet osallistuivat jatkokursseille itsenäisesti työskennelleitä aktiivisemmin. Pariohjelmoinnin ei siis havaittu vaikuttavan heikentävästi opiskelijoiden suoriutumiseen ohjelmoinnista itsenäisesti myöhemmillä opintojaksoilla. (McDowell ym. 2006, 90-95.)

Opiskelijat suhtautuivat parin kanssa työskentelyyn positiivisesti, ja kommunikointi sujui suurimmalla osalla hyvin (Gehringer 2003, 187-190). Enemmistö halusi kokeilla pariohjelmointia myös tulevilla opintojaksoilla (Hanks 2006, 113-117). Ohjelman suunnittelun merkitys korostui, koska opiskelijat ymmärsivät, että ratkaisusta pitää keskustella ja päättää ennen ohjelmoinnin aloittamista. Näin ohjelmista tuli myös toimivampia. Samalla opiskelijat oivalsivat suunnittelun ja yhteistyön edut. Opiskelijat oppivat toisiltaan ja oppimisympäristö oli heidän mielestään avoimempi. (DeClue 2003, 49-56.) Opiskelijat työskentelivät intensiivisemmin parin kanssa. Aikaa ei kulunut sähköpostin lukemiseen tai nettisurffailuun. (Williams 1999, 26-27.) Opiskelijat eivät myöskään halunneet leimautua huonoiksi pareiksi, joten he panostivat opintoihin enemmän kuin yksin työskennellessään. Pariohjelmointia arvostivat erityisesti kokeneemmat ohjelmoinnin opiskelijat (VanDeGrift 2004, 2-6). Oman ohjelmakoodin selittäminen ja parin koodin lukeminen edisti ymmärtämistä ja virheiden huomaamista (Williams & Kessler 2000, 59-65).

Suurin osa opiskelijoista piti parien vaihdosta, koska se edisti ohjelmakoodin dokumentointia. Perusteellisesti kommentoidun ohjelmakoodin sisäisti nopeammin, mikä edisti uuden parin kanssa työskentelyä. (DeClue 2003, 49-56.) Uuden parin myötä oppi aina jotain uutta. Hyvän parin kanssa olisi voinut työskennellä pitempäänkin, mutta toisaalta huonon parin osuessa kohdalle, hänestä pääsi parin vaihdon jälkeen eroon. (Cliburn 2003, 20-29.)

Opiskelijat, jotka eivät olleet tyytyväisiä pariohjelmointiin, antoivat syyksi sen, että he joutuivat tekemään kaiken työn itse tai että aikaa kului koodin selittämiseen parille, joka ei ohjelmakoodia ymmärtänyt. Osa koki, ettei parista ollut heille hyötyä, koska he ovat niin hyviä ohjelmoijia. (Cliburn 2003, 20-29.) Parhaat tulokset saavutettiin, jos parin osaamistaso oli sama tai lähes sama omaan ohjelmointiosaamiseen verrattuna. Parilta sai apua, eikä tarvinnut odottaa opettajan neu-

voja päästäkseen eteenpäin. (Nagappan, Williams, Wiebe, Miller, Balik, Ferzli & Petlick 2003, 185-198.)

Joissakin tutkimuksissa opiskelijat mainitsivat pariohjelmoinnin olleen osittain ajanhukkaa ja työn-
jaon olleen epäselvä. Yksinkertaisissa ohjelmointitehtävissä pari tuntui turhalta. (Muller & Tichy
2001, 537-544.) Parityöskentelylle oli hankalaa löytää yhteistä aikaa oppituntien ulkopuolelta (Cliburn
2003, 20-29). Myös parin erilainen persoona ja taitotaso tuottivat hankaluuksia. Parista ei
ollut hyötyä, jos pari oli heikko. Heikompi pari saattoi tuntea itsensä alempiarvoiseksi edis-
tyneemmän parin rinnalla. (DeClue 2003, 49-56.) Joissakin tapauksissa ongelmien ratkaiseminen
oli aiheuttanut eripuraa parien välille (Cliburn 2003, 20-29).

Myös opetushenkilöstön kokemuksia pariohjelmoinnista on kartoitettu kokeilujen yhteydessä. Op-
pitunnit, joilla työskennellään itsenäisesti, ovat hiljaisia ja välillä turhauttavia opiskelijoille, koska
he joutuvat odottamaan opettajan neuvoja päästäkseen eteenpäin. Opiskelijat saattavat passivoi-
tua, koska apua ei ole välittömästi saatavilla. Usein opiskelijoiden kysymykset ovat yksinkertaisia,
joihin opiskelutoverikin voisi vastata, mutta toveria ei useinkaan häiritä, koska jokainen työstää
omia ratkaisujaan. Yhteistyö on satunnaista. Oppitunnit, joilla harjoitetaan pariohjelmointia, ovat
puolestaan usein äänekkäitä. Opiskelijat auttavat toisiaan ja opettajalle jää enemmän aikaa vas-
tata haasteellisempiin kysymyksiin, koska parit pystyvät ratkaisemaan helpommat ongelmat kes-
kenään. Opettajat ovat tyytyväisempiä työhönsä ja kokevat roolinsa mielekkäämmäksi, koska
heillä on enemmän aikaa paneutua haastavampiin kysymyksiin ja opiskelijat ovat vaikuttaneet
kiinnostuneemmilta oppimaan uutta. (Williams, Wiebe, Yang, Ferzli & Miller 2002, hakupäivä
15.2.2013.)

Opettajan merkitys on ratkaiseva pariohjelmoinnin onnistumisen kannalta. Opettajan pitää antaa
selkeät ohjeet ja huolehtia siitä, että pariohjelmoinnin periaatteita noudatetaan. Rooleja pitää
vaihtaa säännöllisesti ja työskentelyn täytyy olla tasapuolista. (Williams ym. 2002, hakupäivä
15.2.2013.) Tutkimuksissa opettajat kertoivat, että pariohjelmointi vähentää opiskelijoiden turhau-
tumista ja luo kannustavamman ilmapiirin. Opiskelijat oppivat enemmän ja harjoitustöiden laatu
paranee. Pariohjelmointi vähentää myös opettajien korjattavien harjoitustöiden määrän puoleen.
(Williams 2006, 83-85.) Harjoitustyöt palautetaan lisäksi ajallaan (Williams & Kessler 2000, 59-
65). Opettajan oma asennoituminen parityöskentelyyn vaikuttaa opiskelijoiden suhtautumiseen
ratkaisevasti (Hanks 2006, 113-117).

3 TUTKITTAVA OPINTOJAKSO

3.1 Olio-ohjelmoinnin opintojakson esittely

Pariohjelmointikokeilu toteutettiin Oamkin Liiketalouden yksikön Tietojenkäsittelyn koulutusohjelman olio-ohjelmoinnin opintojaksolla syksyllä 2012. Kyseinen opintojakso valittiin tutkimuskohteeksi lähinnä sen ajankohdan, laajuuden ja valinnaisuuden vuoksi. Valintaan vaikutti myös opintojakson opettajan halukkuus osallistua kokeiluun. Kokeilu olisi voitu suorittaa myös jollain toisella kuin olio-ohjelmoinnin opintojaksolla, sillä pariohjelmoinnin käyttäminen työskentelytapana ei ole sidottu tiettyyn ohjelmointikieleen tai ohjelmoinnin lähestymistapaan.

Olio-ohjelmoinnin tarkoituksena on helpottaa ja selkiyttää ohjelmistojen kehittämistä, laajennusta ja ylläpitoa. Perinteisessä ohjelmoinnin lähestymistavassa ohjelmistot koostuvat monimutkaisista listoista tietokonekäskyjä. Olio-ohjelmoinnilla laadittu ohjelma muodostuu itsenäisistä olioista, jotka toimivat yhteistyössä muiden olioiden kanssa. Oliot vastaanottavat, käsittelevät ja lähettävät tietoa toisilensa. Jokainen olio vastaa tietystä tehtävästä ohjelmassa. Ne ovat irrotettavissa ja käytettävissä muissakin kuin alkuperäisessä tarkoituksessaan. Yhteen olioon liittyvä lähdekoodi on yleensä helpommin hahmotettavissa kuin perinteisellä ohjelmointitavalla laaditut käskylistat. (Pohjolainen 2008, hakupäivä 31.3.2013.)

Olio-ohjelmoinnin opintojakso on osa vapaasti valittavia opintoja. Opintojakson laajuus on 6 opintopistettä ja se suositellaan suoritettavaksi toisena tai kolmantena opiskeluvuotena. Esitietovaatimuksena on aiemmin suoritettu opintojakso johdatus olio-ohjelmointiin. (Oamk 2013, hakupäivä 15.2.2013.)

Olio-ohjelmoinnin opintojaksolla opetus toteutetaan kokonaan lähiopetuksena. Oppitunteja on kaksi kertaa viikossa 90 minuuttia kerrallaan yhden lukukauden eli kahden jakson ajan. Opiskelu on itsenäistä sekä opettajajohtoista. Suoritusvaatimuksina ovat harjoitustehtävät, kaksiosainen harjoitustyö sekä kaksi tenttiä. Opintojakson tavoitteena on, että opintojakson suoritettuaan opiskelija hallitsee olio-ohjelmoinnin periaatteet sekä graafisen tapahtumaohjatun käyttöliittymän toteuttamisen Java-ohjelmointikielellä NetBeans-ohjelmistoympäristössä. (Oamk 2013, hakupäivä 15.2.2013, Ojala 17.8.2012, haastattelu.)

Opintojakso on aiempina toteutuskertoina osoittautunut melko haasteelliseksi johtuen lähinnä opiskelijoiden heikosta lähtötasosta. Opintojakson läpäisyprosentti, joka kertoo opintojakson hyväksytysti suorittaneiden prosenttiosuuden, on viime vuosina vaihdellut 30 ja 75 prosentin välillä. (Ojala 17.8.2012, haastattelu.)

3.2 Pariohjelmoinnin toteuttaminen tutkittavalla opintojaksolla

Pariohjelmoinnin periaatteet esiteltiin opiskelijoille heti opintojakson ensimmäisten oppituntien aikana. Tärkeimmät seikat käytiin läpi opinnäytetyön kirjoittajan laatiman PowerPoint-esityksen avulla (ks. liite 1). Lisäksi erilliseen tekstitiedostoon oli koottu yksityiskohtaisempaa tietoa pariohjelmoinnista. Tiedostot tallennettiin opintojakson sähköiselle oppimisalustalle (Moodle) opiskelijoiden käyttöä varten.

Esittelyn jälkeen opiskelijat suorittivat lähtötasotestin, jonka avulla suoritettiin jako pareihin. Parit pyrittiin muodostamaan siten, että ohjelmointitaidoiltaan tasavahvat opiskelijat pääsivät työskentelemään keskenään. Parit työskentelivät pääsääntöisesti niin, että co-pilotin ja driverin rooleja vaihdettiin jokaisen oppitunnin alussa. Näin jokainen sai työskennellä molemmissa rooleissa suunnilleen saman verran. Parien kokoonpanoa vaihdettiin opintojakson puolivälissä. Parityöskentely ei ollut pakollista, vaan jokainen opiskelija sai valita itselleen sopivimman työskentelymuodon. Pariohjelmointia kuitenkin suositeltiin pääasialliseksi työskentelymuodoksi erityisesti opintojakson alussa. Opiskelijat tekivät oppituntien aikana opintojaksoon kuuluvia harjoitustehtäviä parin kanssa. Harjoitustyöt ja tentit kukin suoritti itsenäisesti.

Opintojaksolle varattu opetustila ei kalustejärjestelyiltään ollut sopiva pariohjelmoinnin toteuttamiseen. Yhdelle henkilölle tarkoitetut kulmatyöpisteet ja ahdas tila eivät tarjonneet parasta mahdollista ympäristöä parityöskentelyyn. Tilan vaihtaminen ei ollut mahdollista tilojen varaustilanteen vuoksi.

4 KYSELYT

4.1 Kyselyjen toteuttaminen

Olio-ohjelmoinnin opiskelijoille laadittiin syksyn 2012 ja kevään 2013 aikana yhteensä neljä eri kyselyä (ks. liitteet 2, 3, 4 ja 5) Google Drive -palvelun avulla. Kyselylomakkeen luomisen jälkeen lomake ja vastaustaulukko jaettiin opintojakson opettajalle, jotta hänelläkin oli mahdollisuus tarvittaessa kommentoida kyselylomaketta sekä tarkastella kyselyn tuloksia. Kyselyihin vastattiin pääsääntöisesti oppitunneilla opettajan jakaman verkkolinkin kautta. Kyselyt laadittiin mahdollisimman selkeiksi, yksinkertaisiksi ja lyhyiksi. Näillä toimenpiteillä pyrittiin varmistamaan mahdollisimman suuri vastausten määrä. Kysymykset olivat suurimmaksi osaksi monivalintakysymyksiä, joiden vastausvaihtoehdoista opiskelija valitsi itselleen sopivimman.

Opiskelijat vastasivat ensimmäiseen kyselyyn opintojakson alussa, toiseen kyselyyn opintojakson puolivälissä, kolmanteen kyselyyn opintojakson lopussa ja viimeiseen noin kolme kuukautta opintojakson päättymisen jälkeen. Eri kyselyillä selvitettiin mm. pariohjelmoinnin toteutumista, pariohjelmoinnin vaikutusta oppimiseen, suhtautumista pariohjelmointiin sekä suhtautumisen muuttumista opintojakson edetessä.

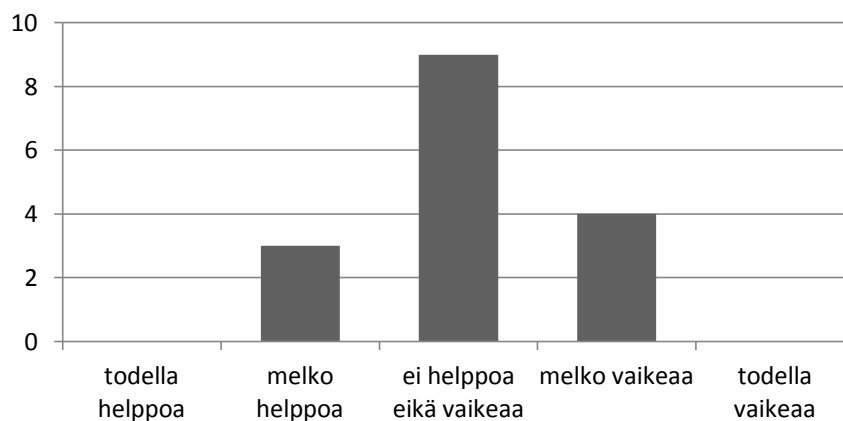
Tutkittavalle opintojaksolle osallistui alun alkaen 16 opiskelijaa. Melko pienen vastaajamäärän vuoksi kyselyiden tuloksista ei voi tehdä laajoja johtopäätöksiä tai yleistyksiä. Kyselyiden tulokset antavat kuitenkin suuntaa pariohjelmoinnin toimivuudesta kyseisellä opintojaksolla.

Opiskelijoille suunnattujen kyselyiden lisäksi opintojakson opettaja vastasi hänelle lähetettyihin avokysymyksiin omien kokemusten ja havaintojen perusteella. Kysymyksillä oli tarkoitus selvittää, miten hyvin pariohjelmointi toimi opintojaksolla opettajan näkökulmasta, miten pariohjelmointi voisi toimia vieläkin paremmin ja oliko pariohjelmoinnilla vaikutusta opiskelijoiden arvosanoihin tai opintojakson läpäisyprosenttiin.

4.2 Ensimmäinen kysely: Pariohjelmoinnin aloitus

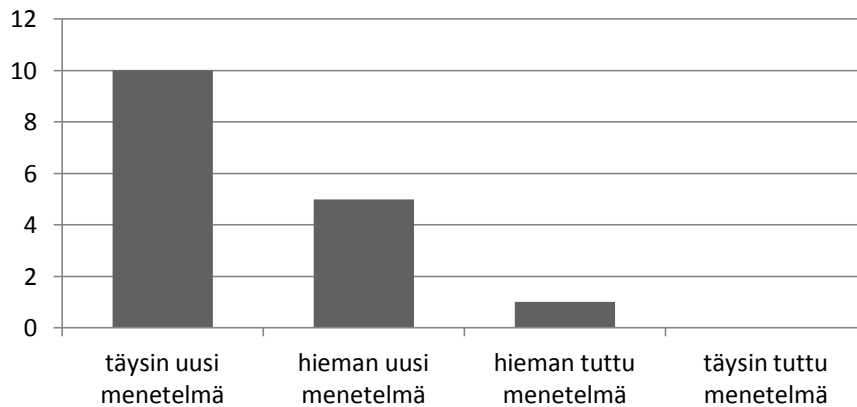
Ensimmäinen kysely (ks. liite 2) toteutettiin opintojakson alussa opintojakson esittelyn, ohjelmointitaitojen alkukartoituksen ja pariohjelmointiin perehdyttämisen jälkeen. Kyselyn tarkoituksena oli selvittää opiskelijoiden taustatietojen lisäksi, kuinka haastavaksi opiskelijat ohjelmoinnin yleensä kokivat ja kuinka tuttua pariohjelmointi oli entuudestaan. Lisäksi kartoitettiin pariohjelmoinnin perehdytysmateriaalin riittävyyttä, ennakkokäsityksiä pariohjelmoinnista sekä parien muodostamisesta opettajajohtoisesti alkukartoituksen pohjalta.

Ensimmäiseen kyselyyn vastasi 16 opiskelijaa. Vastaajista kolme oli naisia. Seitsemän vastaajaa oli iältään 19–21 vuotta, kuusi 22–24 vuotta ja kolme 25 vuotta tai vanhempia. Kaikki vastaajat opiskelivat tietojenkäsittelyä toista vuotta. Yli puolet vastaajista ilmoitti, ettei ohjelmointi yleensä ole heidän mielestään helppoa eikä vaikeaa. Kuviosta 1 käy ilmi, miten haastavana opiskelijat ohjelmoinnin yleensä kokivat.



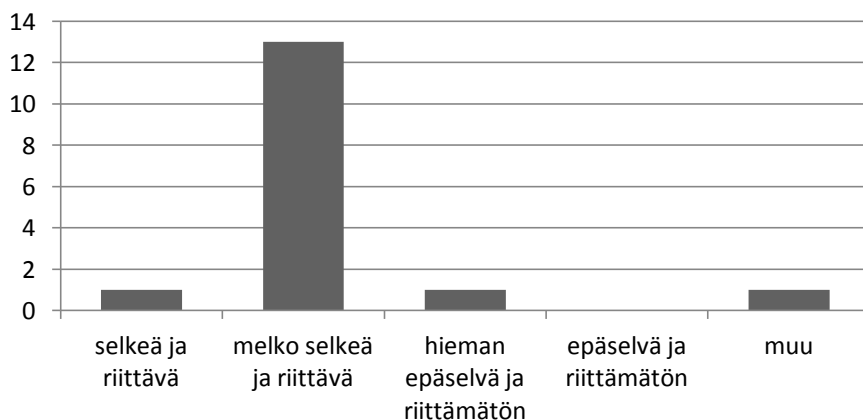
KUVIO 1. Ohjelmoinnin haastavuus yleensä (n=16)

Kymmenelle vastaajalle pariohjelmointi oli täysin uusi ohjelmointimenetelmä. Kenellekään menetelmä ei ollut täysin tuttu. Suurimmalle osalle parin kanssa ohjelmoiminen oli siis uusi kokemus. (ks. kuvio 2).



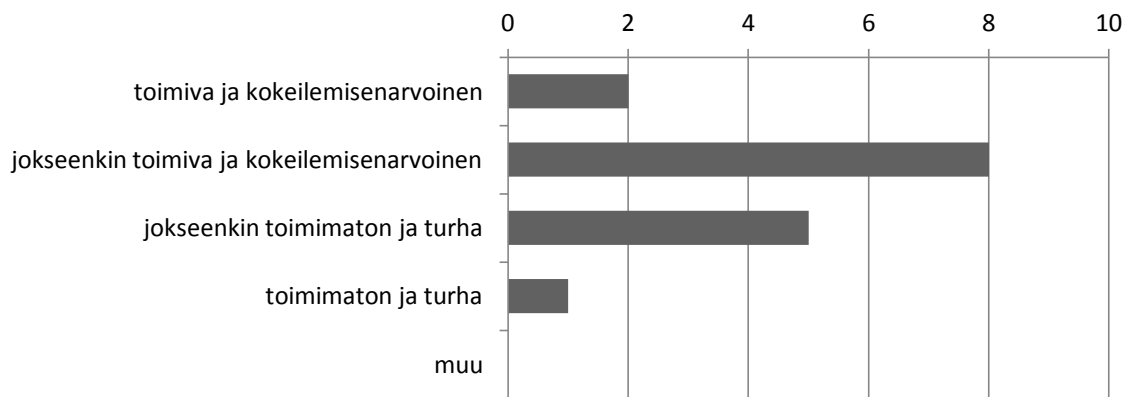
KUVIO 2. Pariohjelmoinnin tuttuus vastaajille (n=16)

Kolmentoista opiskelijan mielestä opintojakson alussa annettu opastus pariohjelmointiin sekä ohjemateriaali olivat melko selkeitä ja riittäviä. Yksi vastaaja ei ottanut kantaa varsinaiseen kysymykseen kommentillaan pariohjelmoinnista yleensä. Kuvioista 3 selviää opiskelijoiden näkemykset ohjeistuksen selkeydestä ja riittävydestä.



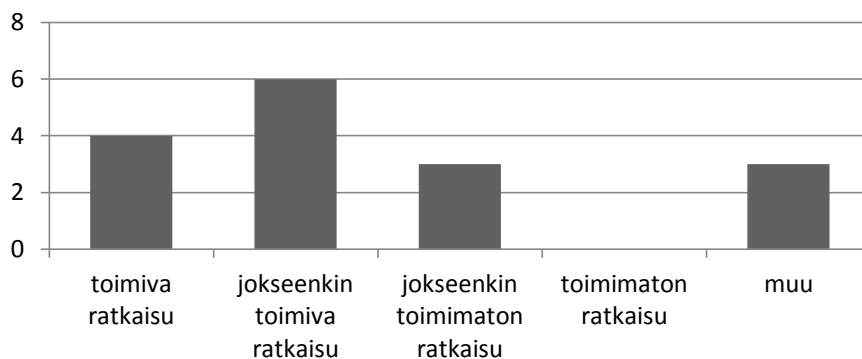
KUVIO 3. Vastaajien mielipide pariohjelmoinnin ohjeistuksen selkeydestä ja riittävydestä (n=16)

Yli puolet vastaajista oli sitä mieltä, että pariohjelmointi vaikutti vähintään jokseenkin toimivalta ja kokeilemisenarvoiselta menetelmältä. Ainoastaan yhden mielipide oli, että pariohjelmointi vaikutti turhalta ja toimimattomalta. Kuvioista 4 käy ilmi ensivaikutelma pariohjelmoinnista.



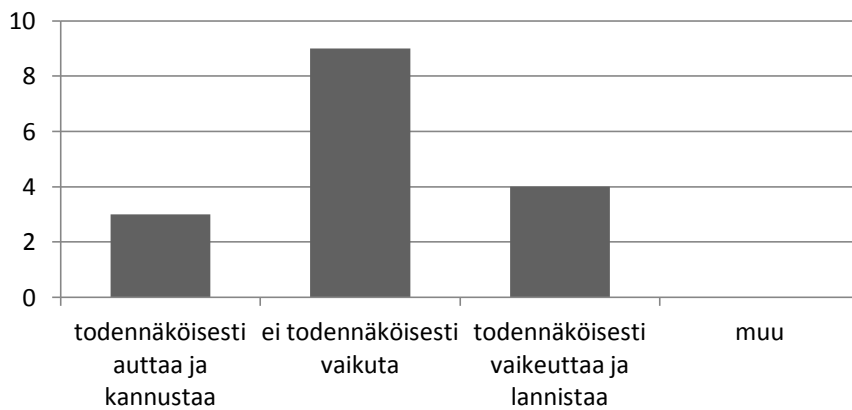
KUVIO 4. Vastaajien ensivaikutelma pariohjelmoinnista (n=16)

Parien muodostus opettajan johdolla alkukartoituksen pohjalta oli kymmenen vastaajan mielestä toimiva tai jokseenkin toimiva ratkaisu. Kolme vastaajaa valitsi vaihtoehdon ”muu”. Kahden vastaajan kommentit olivat hyvin negatiivisia käytettyä menetelmää kohtaan ja yksi ei vielä osannut ottaa asiaan kantaa. Kuviosta 5 selviää vastaajien suhtautuminen tapaan, jolla parit muodostettiin.



KUVIO 5. Vastaajien mielipide parien muodostamisesta opettajajohtoisesti alkukartoituksen pohjalta (n=16)

Yhdeksän opiskelijan näkemys oli, ettei pariohjelmointi todennäköisesti vaikuta lainkaan opintojakson vaatimuksista suoriutumiseen. Vastaajien joukossa oli enemmän sellaisia opiskelijoita, jotka uskoivat pariohjelmoinnin heikentävän mahdollisuuksiaan opintojaksosta suoriutumiseen kuin heitä, joiden mielestä pariohjelmointi todennäköisesti parantaa mahdollisuuksia selvittää opintojaksosta (ks. kuvio 6).



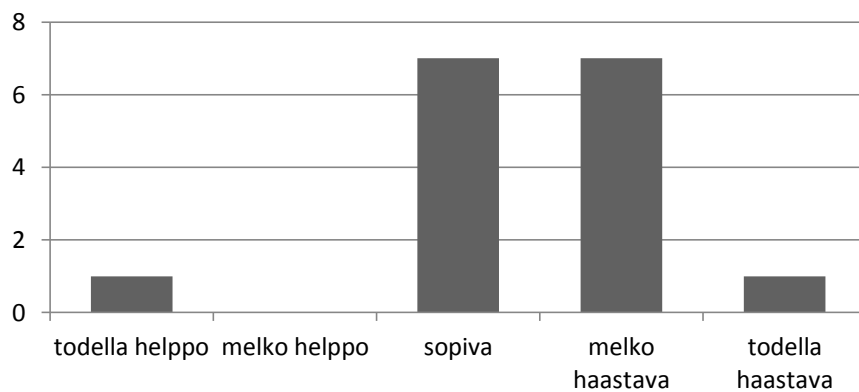
KUVIO 6. Vastaajien arvio parihjelmoinnin vaikutuksesta opintojaksosta suoriutumiseen (n=16)

Ensimmäisen kyselyn viimeisessä kysymyksessä opiskelijat saivat vapaasti esittää näkemyksiä ja kommentoida kyselyä tai parihjelmointia. Kymmenen opiskelijaa oli jättänyt kohdan tyhjäksi. Kuuden vastaajan lisäämät kommentit olivat melko epäileviä parihjelmoinnin ja erityisesti parien muodostuksen suhteen. Eräs opiskelija kommentoi tulevaa kokeilua seuraavasti: ”Aika epäilevin mielin lähdän tähän parihjelmointikokeiluun. Jos parin olisi saanut päättää itse niin ei olisi mitään ongelmaa, mutta jos henkilökemiat eivät toimi niin koko kurssin suorittaminen voi olla varsin ankea kokonaisuus”. Kaikki kommentit kyselyn viimeiseen kohtaan löytyvät koottuna liitteestä 6.

4.3 Toinen kysely: Parihjelmoinnin toteuttaminen

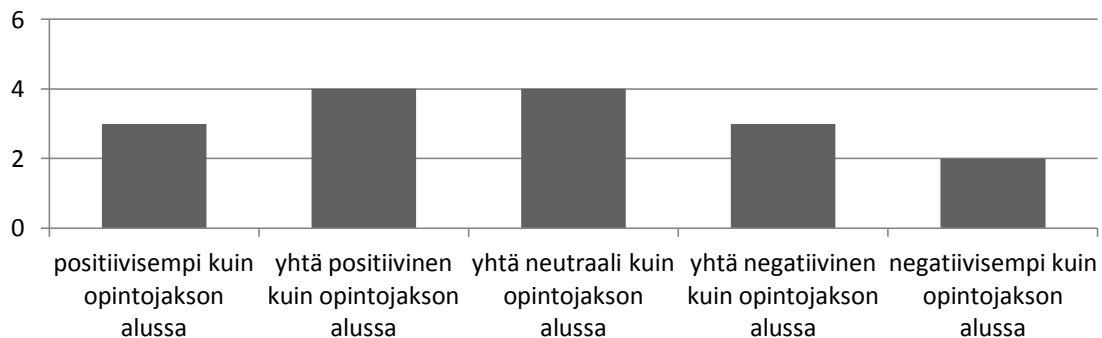
Toinen kysely (ks. liite 3) suoritettiin opintojakson puolivälissä. Kyselyn tarkoituksena oli selvittää olio-ohjelmoinnin opintojakson haastavuutta, suhtautumista parihjelmointiin opintojakson alkuun verrattuna, parihjelmoinnin vaikutusta poissaoloihin ja keskittymiseen oppitunneilla, parityöskentelyn sujuvuutta, syitä parityöskentelyn ylläpitämiseen tai parihjelmoinnista luopumiseen sekä jatkosuunnitelmia työskentelymuodon suhteen opintojakson jälkimmäisellä puoliskolla.

Kyselyyn vastasi 16 opiskelijaa. Opintojaksolle osallistuvien kokoonpanossa oli tapahtunut pieniä muutoksia ensimmäiseen kyselyyn verrattuna, mutta vastaajamäärä pysyi samana. Vastaajista 14 oli miehiä ja kaksi naisia. Vastaajista seitsemän oli 19–21-vuotiaita, kuusi 22–24-vuotiaita ja kolme 25-vuotiaita tai vanhempia. Opiskelijat kokivat olio-ohjelmoinnin opintojakson vaikeustason enimmäkseen sopivaksi tai melko haastavaksi opintojakson puolivälissä (ks. kuvio 7).



KUVIO 7. Olio-ohjelmoinnin opintojakson haasteellisuus (n=16)

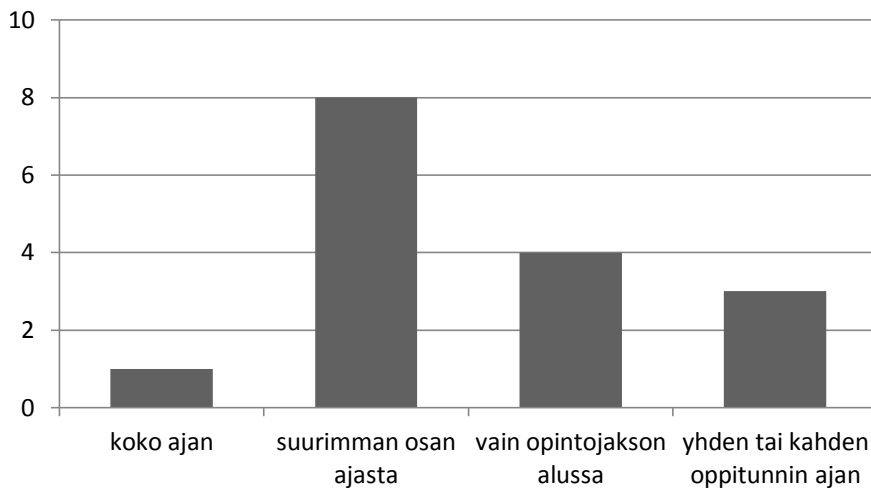
Yhdentoista opiskelijan suhtautuminen pariohjelmointiin ei ollut muuttunut opintojakson alkuun verrattuna. Kolme suhtautui pariohjelmointiin positiivisemmin kuin ensimmäisessä kyselyssä, kun taas kahden vastaajan suhtautuminen oli negatiivisempaa. Opiskelijoiden suhtautuminen pariohjelmointiin opintojakson alkuun verrattuna on esitetty kuviossa 8.



KUVIO 8. Vastaajien suhtautuminen pariohjelmointiin opintojakson alkuun verrattuna (n=16)

Yhtä vastaajaa lukuun ottamatta opiskelijat kokivat, ettei pariohjelmointi ollut vaikuttanut poissaolojen määrään tai keskittymiseen oppitunneilla. Yhden opiskelijan osalta pariohjelmointi oli vähentänyt poissaoloja ja lisännyt keskittymistä.

Yhdeksän opiskelijaa ilmoitti ohjelmoineensa parin kanssa suurimman osan ajasta tai koko ajan. Lähes puolella pariohjelmoinnin hyödyntäminen oli jäänyt kuitenkin hyvin vähäiseksi (ks. kuvio 9).

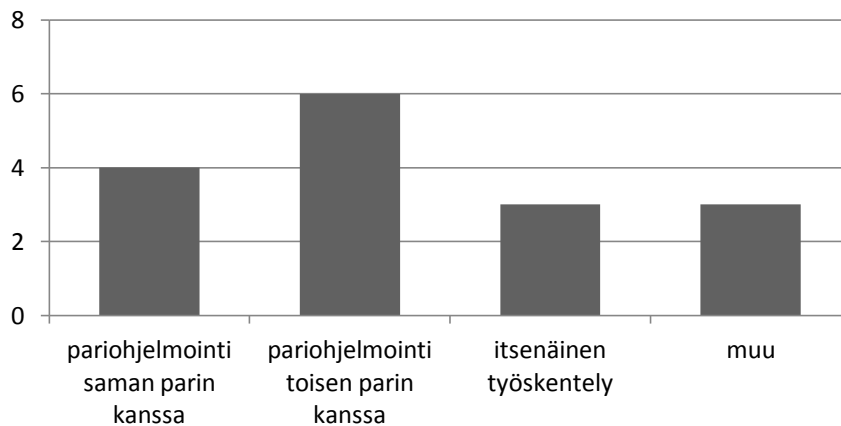


KUVIO 9. Pariohjelmoinnin hyödyntäminen opintojakson alkupuoliskolla (n=16)

Niistä yhdeksästä opiskelijasta, jotka ilmoittivat toteuttaneensa pariohjelmointia joko koko ajan tai suurimman osan ajasta, kolme kertoi tärkeimmäksi syyksi henkilökemioiden toimivuuden. Kolmen vastaajan mielestä tärkein syy pariohjelmointiin oli ohjelmointitaitojen parantuminen. Kolme opiskelijaa valitsi tärkeimmäksi tekijäksi sen, että pariohjelmointi toi vaihtelua itsenäiselle työskentelylle.

Yhteensä seitsemän opiskelijaa vastasi käyttäneensä pariohjelmointia vain opintojakson alussa tai 1-2 oppitunnin ajan. Kahdella vastaajalla syynä oli opintojakson haasteellisuus ja kaksi opiskelijaa ilmoitti syyksi parinmuodostuksen epäonnistumisen. Yksi sanoi syynä olleen puutteet omissa ohjelmointitaidoissa ja yksi vastaaja suhtautui negatiivisesti parityöskentelyyn. Yhden opiskelijan parin poissaolot oppitunneilta olivat muodostuneet esteeksi parityöskentelylle.

Opintojakson jälkimmäisellä puoliskolla kymmenen vastaajaa aikoi jatkaa pariohjelmointia joko saman tai toisen parin kanssa. Muiden osalta tuleva työskentelytapa oli joko itsenäinen tai vielä harkinnassa. Opiskelijoiden jatkosuunnitelmat työskentelytavan suhteen selviävät kuviosta 10.



KUVIO 10. Vastaajien jatkosuunnitelmat työskentelymuodon suhteen opintojakson jälkimmäiselle puoliskolle (n=16)

Viimeisessä kysymyksessä vastaajat saivat vapaasti lisätä ajatuksiaan, näkemyksiään tai kysymyksiään pariohjelmointiin liittyen. Kuusi opiskelijaa kommentoi pariohjelmoinnin toimivuutta opintojaksolla. Eräs opiskelija kommentoi suhtautumistaan pariohjelmointiin näin: "Olin aluksi aika skeptinen koko pariohjelmoinnin suhteen, mutta kun kohdalleni sattui tuttu kaveri joka on suurin piirtein samalla tasolla koodauksen kanssa niin pariohjelmointi on toiminut. Mutta jos pariksi olisi tullut joku hiljaisempi kaveri jonka kanssa ei pääse samalle aaltopituudelle, olisin varmaan tehnyt hommia enemmän yksikseen". Kaikki kommentit ovat luettavissa liitteestä 6.

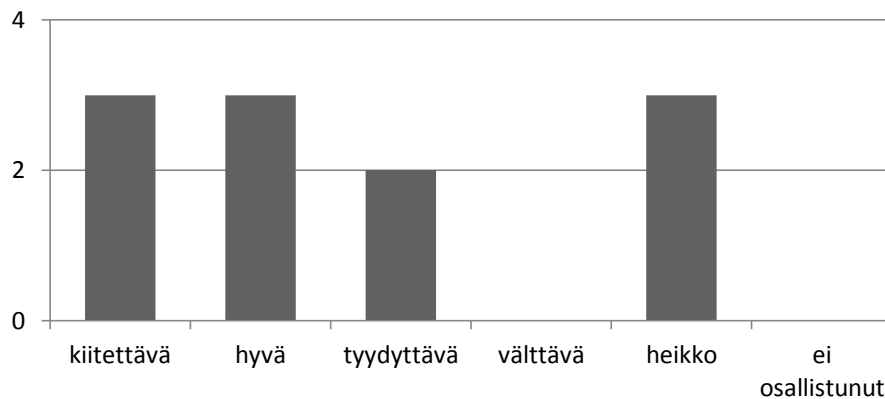
4.4 Kolmas kysely: Pariohjelmoinnin päättäminen

Kolmas kysely (ks. liite 4) opiskelijoille toteutettiin olio-ohjelmoinnin opintojakson päätteeksi toisen tentin yhteydessä, johon osallistui 14 opiskelijaa. Myös tässä kyselyssä oli kymmenen kysymystä. Kysely erosi kuitenkin aiemmista siten, että se sisälsi enemmän avokysymyksiä, joihin opiskelijat kirjoittivat omat vastauksensa valmiiden vaihtoehtojen valitsemisen sijaan.

Kyselyyn vastasi 11 opiskelijaa. Tenttiin osallistuneista ja siten myös kyselylinkin saaneista opiskelijoista kolme jätti vastaamatta. Tämä johtunee siitä, että kyselylle ei varattu tiettyä ajankohtaa osana normaalia oppituntia. Jokainen opiskelija sai vastata kyselyyn tentin jälkeen omalla ajallaan.

Yhdestätoista vastaajasta yksi oli nainen ja loput kymmenen miehiä. Kuusi vastaajaa oli iältään 22–24 vuotta, neljä 25 vuotta tai vanhempi ja yksi 19–21 vuotta.

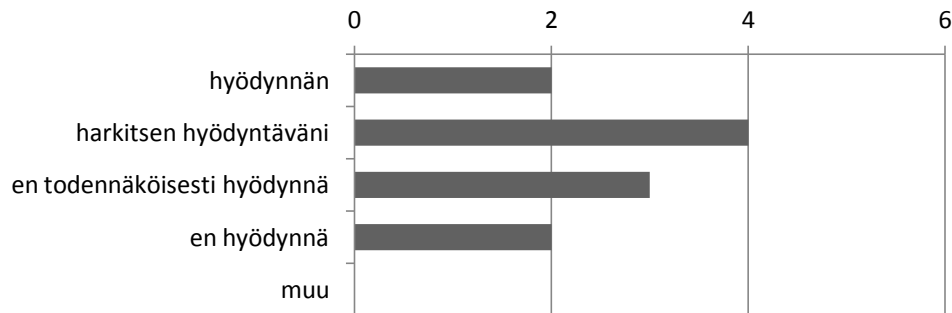
Kolmannen kysymyksen avulla kartoitettiin opiskelijoiden menestymistä opintojakson ensimmäisessä tentissä, joka suoritettiin marraskuun alussa opintojakson puolivälin jälkeen. Kaikki vastaajat olivat osallistuneet tenttiin. Kuviosta 11 käy ilmi opiskelijoiden tenttimenestys. Vastausvaihtoehdoissa ei noudatettu perinteistä arvosanjakaumaa 0-5, vaan opiskelijat antoivat oman subjektiivisen arvionsa menestymisestään.



KUVIO 11. Menestyminen opintojakson ensimmäisessä tentissä (n=11)

Opintojakson jälkimmäisellä puoliskolla vain viisi opiskelijaa ilmoitti ohjelmoineensa parin kanssa suurimman osan ajasta tai koko ajan. Yli puolella pariohjelointi oli ollut satunnaista tai työskentely oli tapahtunut täysin itsenäisesti. Viiden opiskelijan mielestä pariohjelointi oli kannustanut ja auttanut suoriutumaan opintojakson vaatimuksista. Neljän mielestä pariohjelointi ei ollut vaikuttanut opintojaksosta suoriutumiseen. Kahta opiskelijaa pariohjelointi oli lannistanut ja vaikeuttanut heidän suoriutumistaan opintojaksosta.

Kuudennessa kysymyksessä tiedusteltiin, aikovatko opiskelijat hyödyntää pariohjelointia tulevilla ohjelmoinnin opintojaksoilla ja/tai työelämässä. Kuusi opiskelijaa vastasi ainakin harkitsevansa pariohjelmoinnin käyttämistä myöhemminkin. Kuvio 12 havainnollistaa opiskelijoiden vastauksia pariohjelmoinnin hyödyntämisestä jatkossa.



KUVIO 12. Pariohjelmoinnin hyödyntäminen tulevilla opintojaksoilla ja/tai työelämässä (n=11)

Kolmannen kyselyn neljä viimeistä kysymystä olivat avokysymyksiä, joihin opiskelijat laativat omat vastauksensa valmiiden vastausvaihtoehtojen valinnan sijaan. Kaikki vastaukset on luetta-
vissa liitteestä 6. Kysymykset 7 ja 8 kartoittivat syitä, miksi opiskelijat olivat hyödyntäneet parioh-
jelmointia tai miksi eivät. Opiskelijat antoivat mm. seuraavanlaisia perusteluja parin kanssa oh-
jelmoimiseen:

”Pariohjelmointia suositeltiin opintojakson alussa. Pariohjelmoinnin hyödyntäminen vähensi vir-
heiden määrää ja nopeutti niiden löytämistä ohjelmakoodista. Asioista joista oli hieman epävar-
ma, saattoi kysyä mielipidettä ohjelmointiparilta.”

”Mielenkiinto erilaista ohjelmointitapaa kohtaan. Huomioitavaa olisi että parit olisivat kutakuinkin
samalla tasolla.”

Ne opiskelijat, jotka eivät olleet juurikaan ohjelmoineet parin kanssa opintojakson aikana peruste-
livat valintaansa seuraavasti:

”Opintojakson ensimmäisellä puoliskolla en ohjelmoinut parin kanssa suuren tasoeron ja huonoh-
kojen henkilökemioiden takia.”

”Ensimmäinen pari oli pois puolet tunteista. Toisen parin kanssa päätettiin, että työskentelemme
paremmin itsenäisesti, mutta kommunikoimme ja autamme toisiamme omien päätteidemme ääril-
tä.”

Toiseksi viimeisessä kysymyksessä opiskelijoilta pyydettiin kehitysehdotuksia pariohjelmoinnin hyödyntämiseen kyseisellä opintojaksolla. Opiskelijat ehdottivat mm. paritenttejä ja laajempien projektien työstämistä parin kanssa. Pareja haluttiin kierrättää useammin ja pareista toivottiin tasaväkisempiä. Parien muodostamiseen haluttiin vaikuttaa myös itse. Kahdessa vastauksessa pariohjelmoinnin ei katsottu soveltuvan kyseiselle opintojaksolle tai työskentelymuodoksi ylipäänsä.

Jotkut vastaukset kolmannen kyselyn kahdessa viimeisessä kysymyksessä (kysymykset 9 ja 10) antavat aihetta olettaa, että kaikki vastanneet eivät olleet sisäistäneet pariohjelmoinnin työskentelyperiaatteita. Pariohjelmoinnin ei katsottu eroavan satunnaisesta kaverin auttamisesta juurikaan. Lisäksi pariohjelmointia moitittiin opiskelumenetelmänä, koska siinä ei päässyt aina itse tekemään ja ratkaisemaan ongelmia. Vastauksista päätellen osapuolten kommunikointi keskenään ei ollut jatkuvaa ja aktiivista.

4.5 Neljäs kysely: Pariohjelmoinnin hyödyntäminen opintojakson päättymisen jälkeen

Neljäs kysely (ks. liite 5) toteutettiin noin kolme kuukautta opintojakson päättymisen jälkeen kevätlukukaudella 2013. Kyselyn tarkoituksena oli selvittää, olivatko opiskelijat hyödyntäneet pariohjelmointia olio-ohjelmoinnin opintojakson suorittamisen jälkeen. Opiskelijoita pyydettiin myös kertoamaan miten ja miksi he ovat pariohjelmointia käyttäneet. Lisäksi kartoitettiin pariohjelmointiin suhtautumista alkusyksyyn 2012 verrattuna. Opiskelijat saivat myös kertoa mielipiteensä, pitäisikö pariohjelmointia hyödyntää ohjelmoinnin opetuksessa laajemminkin.

Viimeiseen kyselyyn vastasi seitsemän opiskelijaa opintojakson kolmestatoista suorittaneesta opiskelijasta. Vastaajista kuusi oli miehiä ja yksi nainen. Yksi vastaajista oli 19-21-vuotias, kolme 22-24-vuotiaita ja kolme 25-vuotiaita tai vanhempia. Neljä vastaajaa sai opintojaksosta parhaimman arvosanan K5, yksi arvosanan H3 ja kaksi arvosanan T1. Kolme vastaajaa oli hyödyntänyt pariohjelmoinnista olio-ohjelmoinnin opintojakson päättymisen jälkeen. Yksi opiskelija kertoi hyödyntäneensä pariohjelmointia melkein kaikessa ohjelmoinnissa olio-ohjelmoinnin opintojakson päättymisen jälkeen. Kaksi opiskelijaa oli käyttänyt pariohjelmointia työelämälähtöisessä projektissa, jossa isompi ryhmä oli jakautunut pareihin. Yksi opiskelija kertoi jatkaneensa työskentelyä projektissa saman parin kanssa kuin olio-ohjelmoinnin opintojaksollakin (ks. liite 6).

Viiden vastaajan käsitys pariohjelmoinnista oli muuttunut paremmaksi alkusyksyyn 2012 verrattuna. Asennoituminen oli muuttunut positiivisemmaksi, kun opiskelijat olivat huomanneet, kuinka toimivaa ja tehokasta pariohjelmointi on. Osapuolet olivat sekä auttaneet toisiaan että oppineet toisiltaan. Ohjelmakoodin virheet olivat vähentyneet huomattavasti. Opiskelijoiden kommentit löytyvät kokonaisuudessaan liitteestä 6.

Opiskelijoilta kysyttiin myös, pitäisikö pariohjelmointia hyödyntää ohjelmoinnin opetuksessa laajemminkin ja perusteluja heidän mielipiteillensä. Kuuden mielestä pariohjelmointia voisi hyödyntää ainakin jossain määrin, kunhan taidot ovat riittävät, osapuolten taidot vastaavat toisiaan ja molemmilta löytyy halukkuutta yhteistyöhön (ks. liite 6).

4.6 Opettajan haastattelu

Opettajan haastattelu (ks. liite 7) toteutettiin sähköpostin välityksellä olio-ohjelmoinnin opintojakson päätyttyä noin kuukausi ennen kuin arviointi oli saatu lopullisesti päätökseen. Vastauksia täydennettiin ja tarkennettiin vielä arvioinnin valmistuttua kevätlukukaudella 2013.

Opettaja vastasi viiteentoista avokysymykseen opintojaksolta saatujen kokemusten perusteella. Kysymykset koskivat mm. pariohjelmoinnin toimivuutta opetusmenetelmänä, opetustilan soveltuvuutta pariohjelmointiin, parinmuodostuksen onnistumista, opiskelijoiden suhtautumista työskentelymuotoon sekä pariohjelmoinnin vaikutusta arvosanoihin. Syksyn 2012 toteutuskertaa verrattiin myös aikaisempiin toteutuskertoihin opintomenestyksen sekä läpäisyprosentin suhteen.

Olio-ohjelmoinnin opintojakson opettaja kertoi perehtyneensä melko hyvin pariohjelmointiin jo ennen suostumistaan pariohjelmointikokeiluun. Opettajalla ei kuitenkaan ollut aiempaa käytännön kokemusta pariohjelmoinnista työskentely- tai opetusmenetelmänä. Hän kertoi saaneensa lisätietoa ennen opintojakson alkua opinnäytetyön kirjoittajalta saamastaan perehdytysmateriaalista. Opettajan mielestä perehdytysmateriaali oli riittävä, mutta se olisi voinut sisältää ohjeita pariohjelmoinnin toteuttamiseen perinteisessä atk-luokassa, jossa opiskelijat työskentelevät yhdelle opiskelijalle tarkoitettujen kulmapöytien ääressä. Opintojaksolle varattu opetustila ei sopinut kovinkaan hyvin pariohjelmointiin. Tila vaikeutti opiskelijoiden kommunikointia, koska riittävää tilaa vierekkäiselle työskentelylle saman päättien ääressä ei työpisteissä ollut. (Ojala 9.2.2013, haastattelu.)

Opettajan mukaan opiskelijoiden jako tasavahvoihin pareihin alkukartoituksen pohjalta onnistui melko hyvin. Toimivimmat parit muodostuivat opiskelijoista, joilla oli parhaimmat ohjelmointitaidot. Vastaavasti heikoiten parin kanssa työskentelivät ne opiskelijat, joilla oli alhaisin lähtötaso. Opettajan mielestä oli kuitenkin parempi jakaa parit niin, että molempien ohjelmointitaidot olivat samalla tasolla kuin niin, että toinen olisi ollut pariaan merkittävästi parempi. (Ojala 9.2.2013, haastattelu.)

Opiskelijoiden suhtautumista pariohjelmointiin oli opettajan mielestä vaikea arvioida. Opintojakson edetessä keskustelu pariohjelmoinnista väheni opintojakson alkuun verrattuna. Pariohjelmoinnin muuttuminen täysin vapaaehtoiseksi hieman ennen opintojakson puoliväliä ei juuri vaikuttanut parien määrään. Hyvin toimivat parit jatkoivat työskentelyä keskenään kuten aiemminkin. Viimeisinä viikkoina kuitenkin myös nämä parit työskentelivät enemmän itsenäisesti. Opettaja arvioi, että pariohjelmoinnin avulla aktiiviset parit tekivät mahdollisesti lisää eroa ohjelmointitaidoissa itsenäisesti työskennelleisiin. (Ojala 9.2.2013, haastattelu.)

Opettaja koki, että vaativilla opintojaksoilla, joihin olio-ohjelmointi kuuluu, pariohjelmointi toimii, jos opiskelijoilla on riittävän hyvät valmiudet. Pariohjelmointi voisi soveltua paremmin helpommille opintojaksoille, joilla vaatimustaso ei ole niin korkea. Aina on kuitenkin opiskelijoita, jotka eivät pidä parityöskentelystä ja toimivat mieluummin itsenäisesti. (Ojala 9.2.2013, haastattelu.)

Opettajan näkemyksen mukaan suurin syy siihen, ettei pariohjelmointi kaikkien kohdalla toiminut oli kommunikoinnin puutteellisuus. Kommunikointiongelmien taustalla olivat puutteet ohjelmointitaidoissa opintojakson vaatimustasoon nähden. Vaatimustasoa jouduttiin laskemaan, minkä vuoksi kaikkia suunniteltuja asioita ei opintojakson aikana ehditty käsittelemään, koska olio-ohjelmoinnin perusteiden kertaamiseen käytettiin enemmän aikaa kuin aiemmilla toteutuskerroilla. Kaikki opintojakson opetussuunnitelmassa esitetyt asiat ehdittiin kuitenkin toteuttaa. (Ojala 9.2.2013, haastattelu.)

Opiskelijoiden tuntityöskentely oli selkeästi edellisvuotta parempaa ja poissaoloja oli vähemmän. Opettaja ei kuitenkaan osannut arvioida, johtuiko tämä pariohjelmoinnista vai esimerkiksi siitä, että kolmea opintojakson keskeyttänyttä opiskelijaa lukuun ottamatta kaikki opintojaksolle osallistuneet olivat ohjelmointipainotteisen web-sovelluskehityksen suuntautumisvaihtoehdon opiskelijoita. (Ojala 9.2.2013, haastattelu.)

Ensimmäisen tentin ja harjoitustyön perusteella opiskelijat menestyivät opintojaksolla yhtä hyvin tai hieman paremmin kuin aikaisemmillä toteutuskerroilla, mikä käy ilmi opettajan antamista tiedoista. Taulukoissa 2 ja 3 pistekeskisarvot kertovat, kuinka monta prosenttia opiskelijat ovat keskimäärin saavuttaneet harjoitustöiden ja tenttien enimmäispistemäärästä. Ensimmäinen tentti oli syksyllä 2012 poikkeuksellisesti laitetentti, joka on yleensä teorialenttiä vaativampi. Tällä toteutuskerralla perusasioita kerrattiin kuitenkin edellisvuosia enemmän. Toisessa tentissä sekä harjoitustyössä opiskelijat menestyivät syksyllä 2012 keskinkertaisesti tai hieman heikommin aiempiin toteutuskertoihin verrattuna. Pariohjelmointia eniten hyödyntäneet opiskelijat menestyivät opintojaksolla parhaiten. Opettaja arvioi, että pariohjelmointi saattoi auttaa heitä saavuttamaan paremman lopputuloksen. Ne opiskelijat, jotka työskentelivät eniten itsenäisesti, saivat heikompia arvosanoja. (Ojala 9.2.2013, haastattelu.)

TAULUKKO 2. Opiskelijoiden menestyminen olio-ohjelmoinnin opintojakson harjoitustöissä vuosina 2008 - 2012

Ajankohta	Palautetut 1. harjoitustyöt / opiskelijamäärä	1. harjoitustyön pistekeskisarvo	Palautetut 2. harjoitustyöt / opiskelijamäärä	2. harjoitustyön pistekeskisarvo
syksy 2008	22/25	76 %	19/25	55 %
syksy 2009	15/16	83 %	12/16	87 %
syksy 2010	9/11	85 %	8/11	78 %
syksy 2011	13/27	85 %	9/27	68 %
syksy 2012	14/16	85 %	12/16	74 %

TAULUKKO 3. Opiskelijoiden menestyminen olio-ohjelmoinnin opintojakson tenteissä vuosina 2008 - 2012

Ajankohta	1. tenttiin osallistujat / opiskelijamäärä	Ensimmäisen tentin pistekeskisarvo	2. tenttiin osallistujat / opiskelijamäärä	Toisen tentin pistekeskisarvo
syksy 2008	25/25	59 %	25/25	50 %
syksy 2009	15/16	64 %	16/16	70 %
syksy 2010	9/11	69 %	9/11	81 %
syksy 2011	17/27	57 %	13/27	67 %
syksy 2012	15/16	71 %	14/16	53 %

Opintojakson läpäisyprosentti, joka kertoo niiden opiskelijoiden prosenttiosuuden, jotka ovat suorittaneet opintojakson hyväksytysti, oli syksyllä 2012 81 (ks. taulukko 4). Läpäisyprosentti on kor-

kein viimeisten viiden vuoden ajalta. Opettaja ei osannut arvioida, oliko pariohjelmoinnilla vaikutusta opintojakson hyväksytysti suorittaneiden määrään. Selitys voi löytyä myös opiskelijoiden valitsemasta ohjelmointipainotteisesta suuntautumisvaihtoehdosta. (Ojala 9.2.2013, haastattelu.)

TAULUKKO 4. Olio-ohjelmoinnin opintojakson hyväksytysti suorittaneiden osuus vuosina 2008 - 2012

Ajankohta	Aloittaneiden opiskelijoiden lkm	Opintojakson hyväksytysti suorittaneiden lkm	Läpäisyprosentti
syksy 2008	25	19	76 %
syksy 2009	16	12	75 %
syksy 2010	11	8	73 %
syksy 2011	27	9	33 %
syksy 2012	16	13	81 %

Opiskelijoiden saamissa arvosanoissa oli syksyllä 2012 suuria eroja. Kahdeksan opiskelijaa sai parhaan arvosanan K5. Näistä yksi opiskelija korotti alkuperäistä arvosanaansa H3 uusintatentissä. Korkeimman arvosanan saaneiden opiskelijoiden joukossa olivat eniten pariohjelmointia opintojaksolla hyödyntäneet opiskelijat. Heikoimmat arvosanat menivät niille, joilla oli eniten vaikeuksia pariohjelmoinnissa. Arvosanajakaumat vuosilta 2008-2012 käyvät ilmi taulukosta 5. Eri vuosien toteutuskertojen välillä on kuitenkin suuria sisällöllisiä eroja, joten arvosanajakaumat eivät ole suoraan verrannollisia keskenään. (Ojala 9.2.2013, haastattelu.)

TAULUKKO 5. Olio-ohjelmoinnin opintojakson arvosanajakauma vuosina 2008 - 2012

Ajankohta	K5	H4	H3	T2	T1
syksy 2008	4	2	3	6	3
syksy 2009	7	2	2	1	0
syksy 2010	4	3	1	0	0
syksy 2011	2	2	2	3	0
syksy 2012	8	0	2	0	3

Opettajan mielestä on vaikea sanoa, miten pariohjelmointia voisi vielä kehittää olio-ohjelmoinnin opintojaksolla. Syksyn 2012 toteutuskerralla opettaja teki parhaansa, jotta pariohjelmointia hyödynnettäisiin mahdollisimman paljon opetustilan ja opiskelijoiden lähtötason asettamista haasteista huolimatta. Parien muodostaminen tasavahvoista opiskelijoista onnistui melko hyvin, mutta henkilökemioiden ja kommunikointitaitojen pitäisi toimia, jotta pariohjelmointi onnistuu. Opettaja

koki haasteelliseksi saada opiskelijat ymmärtämään, että virheiden teko on sallittua ja jopa suotavaa, jotta pariohjelmointi sujuu. Osa opiskelijoista tuntui arastelevan omien ehdotusten tekemistä parille, jotta ei nolaisi itseään virheiden pelossa. Toiset taas halusivat saada aikaan täydellistä jälkeä, ennen kuin uskalsivat näyttää ohjelmakoodia parilleen. Muutoin opettaja koki, että pariohjelmoinnilla ei ollut vaikutusta opettajan työhön tai rooliin. Erona aikaisempiin toteutuskertoihin oli lähinnä opiskelijoiden kattavampi testaaminen opintojakson alussa taitotason selvittämiseksi parien muodostamista varten. (Ojala 9.2.2013, haastattelu.)

Opettaja kertoi, että voisi käyttää pariohjelmointia työskentelymuotona kaikilla ohjelmoinnin opintojaksoilla, mutta vaativimmilla opintojaksoilla pariohjelmointi olisi vapaaehtoista. Voi olla kuitenkin haasteellista saada opiskelijat suostumaan parityöskentelyyn. Toisena haasteena on opetustilojen soveltumattomuus pariohjelmointiin kalusteiden muodon ja sijoittelun vuoksi. Opettajan mielestä pariohjelmointikokeilu oli tarpeellinen, mutta pilotointia pitää jatkaa, koska kehitettävää löytyy toimivan kokonaisuuden saavuttamiseksi. (Ojala 9.2.2013, haastattelu.)

5 JOHTOPÄÄTÖKSET

Vaikka olio-ohjelmoinnin opintojaksolle osallistuneiden opiskelijoiden ja siten myös pariohjelmointikyselyihin vastanneiden lukumäärä oli suhteellisen pieni, voidaan pariohjelmoinnin soveltuvuudesta opetusmenetelmänä kyseisellä opintojaksolla tehdä kuitenkin johtopäätöksiä. On silti syytä muistaa, että pariohjelmointia pitäisi kokeilla opetusmenetelmänä Oamkin Liiketalouden yksikön Tietojenkäsittelyn koulutusohjelmassa laajemmin, jotta pystyttäisiin tarkemmin kartoittamaan menetelmän hyötyjä ja haasteita ohjelmoinnin opetuksessa.

Tämän tutkimuksen perusteella pariohjelmoinnin toimivuus on ennen kaikkea riippuvainen henkilöiden halukkuudesta ja kyvystä parityöskentelyyn. Myös parien henkilökemioiden kohtaaminen on ensiarvoisen tärkeää. Avoin ilmapiiri sekä opintojaksolle osallistuvien ryhmäyttäminen voisivat edesauttaa pariohjelmoinnin hyödyntämistä opetusmenetelmänä. Pariohjelmointi täytyy esittää opiskelijoille enemmänkin mahdollisuutena kuin pakkona.

Parien muodostuksessa voisi tämän kokeilun ja aiempien tutkimustulosten perusteella ottaa huomioon myös opiskelijoiden omat toiveet. Pareja voisi myös kierrättää hieman enemmän. Parien kokoonpanoissa voisi kokeilla erilaisia vaihtoehtoja. Tässä kokeilussa osapuolten ohjelmointitaidot olivat samalla tasolla, mutta parit voisi muodostaa myös niin, että toinen on ohjelmointitaidoiltaan pariaan vahvempi. Tässä tapauksessa pitää olla kuitenkin tarkkana, ettei lähtötasoltaan parempi ohjelmoija hoida ohjelmointia itsenäisesti heikomman seurattessa passiivisena vieressä.

Opiskelijoiden lähtötasolla on suuri merkitys pariohjelmoinnin toimivuuteen. Yhteistyö on hankalaa, jos ohjelmointitaidoissa on paljon paikattavaa. Ohjelmoinnin opintojaksojen vaatimustasoa voisi ehkä nostaa opiskelujen alkuvaiheessa, jotta kaikilla opiskelijoilla olisi jatkossa paremmat valmiudet selviytyä haastavammista opintojaksoista ja hyödyntää pariohjelmointia. Aiemmissa tutkimustuloksissa ei opiskelijoiden heikkoa lähtötasoa ole laajemmin koettu ongelmaksi, vaikka parien ohjelmointitaidoissa onkin ollut eroja. Pienessä otannassa erot saattavat kuitenkin korostua.

Pariohjelmointi auttaa erityisesti monimutkaisten ohjelmointiongelmien ratkaisemisessa. Tämä huomioiden pariohjelmoinnin pitäisi soveltua työskentelymuodoksi erityisesti vaativammille ohjel-

moinnin opintojaksoille, jollaiseksi olio-ohjelmoinnin opintojakso pääsääntöisesti vastaajien keskuudessa koettiin. Jos opiskelijoiden taidoissa on kuitenkin merkittäviä puutteita, ei pariohjelmointi ole kovinkaan hedelmällistä. Pariohjelmointi vaativimmilla opintojaksoilla edellyttää, että ohjelmoinnin perustaidot ovat hallussa. Ohjelmoinnin peruskursseilla taitotasoissa ei ehkä vielä ole niin suuria eroja kuin jatkokursseilla, joten pariohjelmointia voisi tästä syystä soveltaa jo opiskelun alkuvaiheessa, vaikka ratkaistavat ohjelmointiongelmien eivät olisikaan kovin monimutkaisia.

Tämän kokeilun perusteella vaikuttaisi, että sosiaaliset opiskelijat, joilla ohjelmoinnin perusperiaatteet ovat hyvin hallussa, hyötyvät pariohjelmoinnista kaikkein eniten. He mahdollisesti jopa kasvattavat eroa heikompiin opiskelijoihin nähden, jos he saavat työskennellä parin kanssa, jolla on vastaavat ohjelmointitaidot. Passiiviset opiskelijat, joilla on puutteita ohjelmointitaidoissa, eivät hyödy pariohjelmoinnista. Parityöskentely voi jopa lisätä passivoitumista, mistä on myös aiempaa tutkimusnäyttöä. Opiskelijoiden totuttaminen parityöskentelyyn jo ohjelmoinnin peruskursseilla voisi kenties edistää itsenäistä työskentelyä suosivien asennoitumista pariohjelmointiin. Työelämässä hyvän ohjelmoijan pitää olla aina valmis yhteistyöhön ja jakamaan osaamistaan myös muille. Pariohjelmointiin tottuminen jo opiskeluaikana voi edistää yhteistyökykyä myöhemmin työelämään siirryttäessä.

Tässä opinnäytetyössä toteutetussa pariohjelmointikokeilussa opiskelijat työstivät kahta laajempaa harjoitustyötä itsenäisesti. Aiempien tutkimusten mukaan juuri isommissa töissä saavutetaan parhain hyöty pariohjelmoinnilla. Samalla myös opettajan työmäärä pienenee, koska korjattavia töitä on vähemmän ja niiden laatu on parempi. Tästä syystä olisi hyvä harkita myös harjoitustöiden toteuttamista parin kanssa tulevilla toteutuskertoilla. Myös kyselyihin vastanneet opiskelijat ehdottivat tätä. Tässä tutkimuksessa opettaja ei havainnut juurikaan eroa opettajan roolissa tai työmäärässä aikaisempiin toteutuskertoihin verrattuna lähtötasotestejä lukuun ottamatta. Pienessä ryhmässä opettajalla riittääkin varmasti paremmin aikaa opiskelijoiden kysymyksille, toisin kuin aiempien tutkimusten suurissa opetusryhmissä, joissa oli havaittavissa turhautumista, kun apua ei ollut saatavilla. Opettaja mainitsi kuitenkin, että parin kanssa ohjelmoivien opiskelijoiden toimintaan ei juuri tarvinnut puuttua. Myös opiskelijoiden kommentit, että pariohjelmointi auttoi pättitilanteissa, tukevat opettajan havaintoa.

Aiempien tutkimustulosten mukaan pariohjelmointi ei juuri vaikuta tenttien arvosanoihin, mutta edistää opiskelijoiden osallistumista tentteihin ja opintojaksosta suoriutumista. Tässä opinnäytetyössä tehdyssä tutkimuksessa tutkittavana olleen opintojakson läpäisyprosentti oli korkeampi

kuin aiemmilla toteutuskertoilla vuosina 2008-2011. Tenttimenestys oli ensimmäisessä tentissä aiempia vuosia parempi, mutta toisessa tentissä alhaisin vuosiin 2008-2011 verrattuna. Arvosanjakauma poikkesi aikaisempiin vuosiin verrattuna siten, että korkeimpia arvosanoja oli runsaasti, ja arvosanoja H4 ja T2 ei saatu lainkaan. On mahdollista, että pariohjelmoinnilla on ollut vaikutusta korkeaan läpäisyprosenttiin ja arvosanjakaumaan, mutta myös opiskelijoiden valitsema ohjelmointipainotteinen suuntautumisvaihtoehto voi selittää lukemia.

Opettajan pitää varautua siihen, että voi olla haasteellista saada kaikki opiskelijat suostumaan pariohjelmointiin. Tässä opinnäytetyössä suoritettun kokeilun myötä kävi kuitenkin ilmi, että opiskelijoiden suhtautuminen pariohjelmointiin muuttui positiivisemmaksi opintojakson edetessä, kun opiskelijat huomasivat hyötывänsä pariohjelmoinnista. Se, että pariohjelmointia on hyödynnetty myös opintojakson päättymisen jälkeen, vaikka aluksi suhtautuminen olikin ehkä nihkeää, antaa vahvistusta ajatukselle, että pariohjelmointi toimii opetusmenetelmänä. Opettajan omalla asenteella ja parityöskentelyn seuraamisella on vaikutusta myös opiskelijoiden asennoitumiseen.

Osa kyselyihin vastanneista opiskelijoista ei heidän kommenttinsa perusteella oikein oivaltanut, mitä pariohjelmointi oikein tarkoittaa. Epäselvä roolijako on aiheuttanut haasteita myös aiemmissa tutkimuksissa. Co-pilotin roolia ja merkitystä ei aina sisäistetty, sillä se koettiin rooliksi, jossa oppii vähemmän. Co-pilot ei kuitenkaan ole passiivinen sivustaseuraaja, vaan ehkä jopa driveria tärkeämpi osapuoli, sillä hänen vastuullaan on korjata virheitä, ehdottaa ratkaisuja ohjelmakehitykseen ja olla jatkuvassa vuorovaikutuksessa driverin kanssa. Pariohjelmointia ei myöskään ole se, että molemmat tuottavat omaa lähdekoodia oman päättöen ääressä ja vasta ongelmien ilmaantuessa käännytään toisen puoleen. Roolien merkitystä ja pariohjelmoinnin peruseriaatteita voi joutua perustelemaan ja selkiyttämään useita kertoa, ennen kuin opiskelijat ne omaksuvat.

Opetustilojen tulisi tarjota mahdollisuus parityöskentelyyn. Yhdelle opiskelijalle tarkoitetut työpisetteet aiheuttavat turhan fyysisen esteen pariohjelmoinnin toteuttamiseen. Opetustilojen kalustuksessa pitäisikin kiinnittää enemmän huomiota kalustejärjestelyjen muokattavuuteen tarpeen mukaan. Työskentelypisteet pitäisi sijoittaa ensisijaisesti vierekkäin eikä peräkkäin. Tässä opinnäytetyössä kuvatussa kokeilussa opetustila sopi erittäin huonosti parityöskentelyyn. Tilan vaihtaminen ei kuitenkaan ollut mahdollista, koska tilavaraukset tehdään paljon ennen opetuksen alkua, eikä vapaita pariohjelmointiin sopivia opetustiloja juuri ole.

Pariohjelmoinnin hyödyntämisessä opetuksessa on omat haasteensa, mutta tehdyn kokeilun tulokset ovat kuitenkin kannustavia. Ainakin osa opiskelijoista selvästi hyötyy pariohjelmoinnista. Edellytyksenä kuitenkin on, että opiskelijoiden lähtötaso, halukkuus ja kyky yhteistyöhön sekä opetustila ovat kunnossa. Seuraavaan luetteloon on koottu lyhyesti ohjeita, joita ohjelmoinnin opettaja voi halutessaan noudattaa toteuttaessaan pariohjelmointia omilla opintojaksoillaan.

1. Tutustu pariohjelmoinnin työskentelyperiaatteisiin huolellisesti ennen ensimmäistä pariohjelmointikokeilua. Kokeile pariohjelmointia itse, jos se on mahdollista.
2. Varmista, että opetustila soveltuu pariohjelmointiin (tilaa vierekkäiselle työskentelylle).
3. Varaa riittävästi aikaa opiskelijoiden opastukseen ennen pariohjelmoinnin alkua.
4. Pyri luomaan opetustilaan virheet salliva ja avoin ilmapiiri.
5. Varaudu opiskelijoiden skeptiseen asenteeseen. Mieti perustelut pariohjelmoinnin käyttämiseen ennakkoon. Myös oma asenteesi vaikuttaa opiskelijoiden suhtautumiseen.
6. Pyri jakamaan parit niin, että parin osapuolet ovat tasavahvoja ohjelmointitaidoiltaan (lähtötasotesti tai aikaisemmat arvosanat), jotta opiskelijat eivät passivoidu ja turhaudu.
7. Anna opiskelijoille mahdollisuus vaikuttaa parien muodostamiseen.
8. Parityöskentelyn tulisi olla ensisijainen työskentelymuoto, mutta ei pakollinen.
9. Seuraa parien työskentelyä. Älä anna toisen osapuolen dominoida. Puutu epäkohtiin nopeasti. Roolien omaksuminen voi olla opiskelijoille haasteellista.
10. Rooleja (driver ja co-pilot) pitää vaihtaa riittävän usein (esim. 15-30 minuutin välein tehtävän mukaan), jotta molemmat roolit tuntuvat mielekkäiltä.
11. Pareja kannattaa kierrättää muutaman viikon välein. Pareille pitää kuitenkin jäädä aikaa löytää oma työskentelyrytmi ennen parin vaihtamista.
12. Pariohjelmointia ei kannata käyttää kaikkein yksinkertaisimpiin ohjelmointiharjoituksiin, vaan haastavampiin ja pitkäkestoisiin tehtäviin.
13. Harjoitustöiden teettäminen parin kanssa voi vähentää opettajan työtaakkaa. Myös paritenttejä voisi kokeilla.
14. Työskentelystä parin kanssa kannattaa kerätä palautetta. Opiskelijat voivat antaa toisilleen myös vertaispalautetta tai arvioida omaa panostaan parityöskentelyssä.
15. Jaa kokemuksiasi toisille opettajille, jotta saadaan aikaan toimiva kokonaisuus.

6 POHDINTA

Tämän opinnäytetyön tekeminen on ollut hyvin antoisa ja mielenkiintoinen kokemus. Yhteistyö olio-ohjelmoinnin opintojakson opettajan kanssa sujui luontevasti, vaikka joitain vähäisiä näkemyseroja sekä tiedonkulkukatkoksia matkan varrelle mahtuikin johtuen osittain siitä, että opinnäytetyö on laadittu toiselta paikkakunnalta käsin. Kaikki kommunikaatio tapahtui käytännössä verkon välityksellä. Opinnäytetyön ohjaaja oli eri henkilö kuin tutkittavan opintojakson opettaja, mikä oli mielestäni hyvä ratkaisu, sillä näin tutkimuksen toteuttaminen vastasi enemmän todellista tutkimustilannetta. Ohjaaja antoi arvokkaita ehdotuksia opinnäytetyön toteuttamiseen ja raportointiin. Työstin kyselyitä sekä opinnäytetyötä kuitenkin hyvin pitkälle itsenäisesti. Uskon silti saaneeni kerättyä tarvittavat tiedot, vaikka viimeisteltävää esimerkiksi kyselyihin varmasti jäikin. Kyselyissä olisi voinut lisäksi tiedustella, miten opiskelijat itse mielestään onnistuivat parityöskentelyssä ja parin tukemisessa.

Opinnäytetyöprosessin myötä opin paljon kyselytutkimuksen toteuttamisesta ja tilastollisen tutkimuksen raportoinnista. Pariohjelmointi oli minulle tuttu työskentelymenetelmänä ammattiharjoittelun ansiosta, mutta pariohjelmoinnista opetusmenetelmänä minulla ei ollut aiempaa tietoa. Opinnäytetyön ansiosta voin nyt hyödyntää pariohjelmointia mahdollisesti myös omassa opetustyössä tulevaisuudessa.

Pariohjelmointia on tutkittu enimmäkseen oppilaitosympäristössä, mikä voi osittain antaa vääristyneen kuvan pariohjelmoinnin toimivuudesta työelämässä. Olisikin mielenkiintoista tutkia pariohjelmoinnin etuja ja haasteita myös jossain ohjelmistoyrityksessä. Olosuhteista johtuen en pystynyt tarkkailemaan ja havainnoimaan opiskelijoiden työskentelyä parin kanssa syksyllä 2012. Tämä olisi varmasti antanut lisää tietoa pariohjelmoinnin hyödyntämisestä ja toimivuudesta. Uskon kuitenkin opettajan tekemien havaintojen olleen aitoja ja oikeita. Tutkimukseen olisi voitu saada lisäarvoa, jos olisi ollut mahdollista tutkia kahta rinnakkain etenevää olio-ohjelmoinnin opintojakson toteutuskertaa siten, että toisessa ryhmässä olisi käytetty pariohjelmointia opetusmenetelmänä ja toisessa opetus olisi toteutettu perinteisellä tavalla. Tämä ei kuitenkaan ollut mahdollista, sillä opintojakso toteutetaan vain kerran lukuvuodessa.

Olio-ohjelmoinnin opintojaksolle osallistuneiden ja siten myös kyselyihin vastanneiden määrä oli liian pieni, jotta esimerkiksi opiskelijoiden taustoista ja heidän vastauksistaan olisi voinut tehdä johtopäätöksiä. Olisi ollut mielenkiintoista selvittää, vaikuttaako esim. vastaajan sukupuoli, ikä tai aikaisempi ohjelmointikokemus hänen asennoitumiseensa parityöskentelyyn.

Koen, että sain hyödynnettyä opinnäytetyössäni omia kokemuksiani niin pariohjelmoinnista kuin opettamisestakin. Uskon, että tutkimustuloksista on hyötyä opettajille, jotka harkitsevat pariohjelmoinnin toteuttamista omalla opintojaksollaan. On kuitenkin selvää, että pariohjelmointikokeiluja tarvitaan Oamkissa lisää, jotta käytänteet hioutuvat toimivaksi kokonaisuudeksi.

Omien kokemusteni mukaan pariohjelmointi on hyvin antoisa työskentelymuoto. Toisaalta pariohjelmointi on myös haastavaa ja ajoittain rankkaakin. Pariohjelmointi vaatii kompromisseja ja molempien osapuolten pitää olla avoimia ja sitoutuneita työhön. Ohjelmointityö voi olla kuitenkin ajoittain hyvin yksinäistä puurtamista. Sosiaalisena ihmisenä koen, että pariohjelmointi tuo vaihtelua päivärytmiin ja voi myös edistää työssä jaksamista. Kannustankin Oamkin Liiketalouden yksikön Tietojenkäsittelyn koulutusohjelman opettajia kokeilemaan pariohjelmointia ohjelmoinnin opintojaksoillaan sekä jakamaan kokemuksiaan ja parhaita käytänteitä toisillensa. Ilokseni tietooni on tullut ainakin yksi Oamkissa pariohjelmointiin liittyvä jatkokokeilu, jossa parit muodostetaan toisin kuin tässä opinnäytetyössä kuvaillussa kokeilussa. Tämä kokeilu on tarkoitus toteuttaa vielä lukuvuoden 2013 aikana. Olisi mielenkiintoista kuulla kyseisen kokeilun tuloksista. Se, että pariohjelmointi on aiheuttanut kiinnostusta laajemminkin ainakin osittain tämän opinnäytetyön ansiosta luo uskoa siihen, että tässä opinnäytetyössä kuvailtu kokeilu on ollut tarpeellinen ja että se on antanut uusia toteutusmahdollisuuksia ohjelmoinnin opintojaksoille.

LÄHTEET

Agile Coaching Experience. (Julkaisuvuosi puuttuu). Desks for Pair Programming. Hakupäivä 30.3.2013 <http://agilexp.com/informativeworkspace/desks.php>.

Beck K. 1999. Embracing change with Extreme programming. IEEE Computer 32 (10), 70-77.

Beck, K. & Anders, C. 2004. Extreme Programming Explained: Embrace Change—Second Edition. Addison-Wesley Longman. Boston.

Begel A. & Nagappan N. 2008. Pair programming: What's in it for me?. 2nd International Symposium on Empirical Software Engineering and Measurement (ESEM '08), 120-128.

Cliburn, D. C. 2003. Experiences with pair programming at a small college. Journal of Computing Sciences in Colleges 19 (1), 20-29.

DeClue, T. H. 2003. Pair programming and pair trading: Effects on learning and motivation in a CS2 course. Journal of Computing Sciences in Colleges 18 (5), 49-56.

Cockburn, A. & Williams, L. 2000. The Costs and Benefits of Pair Programming. In 1st International Conference on Extreme Programming and Flexible Processes in Software Engineering, June 21 - 23, Cagliari, Sardinia, Italy. The Pennsylvania State University.

Gehring, E. F. 2003. A pair programming experiment in a non-programming course. Companion of the 18th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages and Applications (OOPSLA '03), Anaheim, CA, USA, 187-190.

Hanks, B. 2006. Student attitudes toward pair programming. Proceedings of the 11th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education (ITICSE '06), Bologna, Italy, 113-117.

Hulkko, H. & Abrahamsson, P. 2005. A Multiple Case Study on the Impact of Pair Programming on Product Quality International Conference on Software engineering St. Louis, MO, USA May 15-21 2005, 495-504.

Katira, N., Williams, L. & Osborne, J. 2005. Towards increasing the compatibility of student pair programmers. Proceedings of the 27th International Conference on Software Engineering (ICSE '05), St. Louis, MO, USA, 625-626.

Katira, N., Williams, L., Wiebe, E., Miller, C., Balik, S. & Gehringer, E. 2004. On understanding compatibility of student pair programmers. Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education (SIGCSE '04), Norfolk, Virginia, USA, 7-11.

Keefe, K., Sheard, J. & Dick, M. 2006. Adopting XP practices for teaching object oriented programming. Proceedings of the 8th Australian Conference on Computing Education (ACE '06), Hobart, Australia, 91-100.

Lui, K. M. & Chan, K. C. C. 2006. Pair programming productivity: Novice-novice vs. expert-expert. International Journal of Human-computer Studies 64 (9), 915-925.

McDowell, C., Werner, L., Bullock, H. & Fernald, J. 2006. Pair programming improves student retention, confidence and program quality. Communications of the ACM 49 (8), 90-95.

Mendes, E., Al-Fakhri, L. B. & Luxton-Reilly, A. 2005. Investigating pair programming in a 2nd-year software development and design computer science course. Proceedings of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education (ITICSE '05), Caparica, Portugal, 296-300.

Muller, M. M. & Tichy, W. F. 2001. Case study: Extreme programming in a university environment. Proceedings of the 23rd International Conference on Software Engineering (ICSE '01), Toronto, Ontario, Canada, 537-544.

Nagappan, N., Williams, L., Ferzli, M., Wiebe, E., Yang, K., Miller, C. & Balik, S. 2003. Improving the CS1 experience with pair programming. Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education (SIGCSE '03), Reno, Nevada, USA, 359-362.

Nagappan, N., Williams, L., Wiebe, E., Miller, C. & Balik, S., Ferzli, M. & Petlick, J. 2003. Pair learning: With an eye toward future success. Proceedings of XP Agile Universe (XP/AU), Springer LNCS 2753, 185-198.

Nawrocki, J. & Wojciechowski, A. 2001. Experimental Evaluation of Pair Programming. In 12th European Software Control and Metrics Conference, April 2-4 2001, London, UK.

Oamk. 2013. Koulutusohjelmat 2012-2013. Opintojaksokuvaus. Olio-ohjelmointi 6 op. Hakupäivä 15.2.2013 http://www.oamk.fi/koulutus_ja_hakeminen/opiskelu_oamkissa/opinto-opas/koulutusohjelmat/?sivu=oj_kuvaus&koodi1=K1034TK&kieli=FI&opas=2012-2013&lk=s2012&vuosi=12S13K.

Ojala, P., lehtori, Oulun seudun ammattikorkeakoulu, Liiketalouden yksikkö. 2012. Haastattelu 17.8.2012. Tekijän hallussa.

Ojala, P., lehtori, Oulun seudun ammattikorkeakoulu, Liiketalouden yksikkö. 2012. Haastattelu 9.2.2013. Tekijän hallussa.

Pohjolainen, J. 2008. Johdatus olio-ohjelmointiin. Hakupäivä 31.3.2013 <http://www.slideshare.net/pohjus/johdatus-olioohjelmointiin-presentation>.

Salo O. & Abrahamsson P. 2008. Agile methods in European embedded software development organizations: A survey on the actual use and usefulness of Extreme Programming and Scrum. IET Software 2 (1), 58-64.

Schindler C. 2008. Agile software development methods and practices in Austrian IT-industry: results of an empirical study. Computational Intelligence for Modelling, Control and Automation (CIMCA '08), 321-326.

Sfetsos, P., Stamelos I., Angelis, L. & Deligiannis I. 2006. Investigating the impact of personality types on communication and collaboration-viability in pair programming, an empirical study. XP/Agile conference 7, 43-52.

Sharifabdi, K. & Grot C. 2002. Team Development and Pair Programming - Tasks and Challenges of the XP Coach. Proceedings of XP2002, Sardinia, Italy, May 26-29, 166-169.

Succi, G. & Marchese, M. 2001. Extreme Programming Examined. Addison Wesley, Boston, 221-320.

Thomas, L., Ratcliffe, M. & Robertson A. 2003. Code warriors and code-a-phobes: A study in attitude and pair programming. SIGCSE '03: Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education, Reno, Nevada, USA, 363-367.

VanDeGrift, T. 2004. Coupling pair programming and writing: Learning about students' perceptions and processes. Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education (SIGCSE '04), Norfolk, Virginia, USA, 2-6.

Van Toll III, T., Lee, R. & Ahlswede, T. 2007. Evaluating the usefulness of pair programming in a classroom setting. Computer and Information Science. 6th IEEE/ACIS International Conference (ICIS 2007), 302-308.

Williams, L. 1999. But, isn't that cheating? [collaborative programming]. Frontiers in 29th Annual Education Conference (FIE '99), 2 (12B9), 26-27.

Williams L. 2001. Integrating Pair Programming into a Software Development Process. In 14th Conference on Software Engineering Education and Training, February 19 - 21, Charlotte, USA.

Williams L. 2006. Debunking the nerd stereotype with pair programming. IEEE Computer Society 39 (5), 83-85.

Williams L. & Kessler R. 2000. The effects of "pair-pressure" and "pair-learning". Proceedings of the 13th Conference on Software Engineering Education & Training, 59-65.

Williams L. & Kessler R. 2003. Pair programming illuminated. Addison-Wesley.

Williams L., Kessler R., Cunningham W. & Jeffries R. 2000. Strengthening the Case for Pair Programming. IEEE Software 17 (4), 19-25.

Williams, L., Wiebe, E., Yang, K., Ferzli, M. & Miller C. 2002. In support of pair programming in the introductory computer science course. Hakupäivä 15.2.2013

http://collaboration.csc.ncsu.edu/laurie/Papers/PP%20in%20Introductory_CSED.pdf.



Pariohjelmointi



Annika Alakstari 31.3.2013

Mitä pariohjelmointi on?

- Ohjelmointimenetelmä ketterässä ohjelmointi-kehityksessä
- Kaksi ohjelmoijaa tekee tiivistä yhteistyötä
- Toinen ohjelmoijista (driver) on näppäimistön ääressä ja toinen toimii tarkkailijana (copilot)
- Rooleja ja pareja vaihdetaan tarvittaessa tai sovituin väliajoin
- Driver tuottaa ohjelmakoodia
- Copilot korjaa logiikka- ja syntaksivirheitä sekä tekee ohjelmakehitykseen liittyviä ehdotuksia
- Jatkuva vuorovaikutusta driverin ja copilotin välillä
- Soveltuu parhaiten monimutkaisiin ongelmiin

Pariohjelmoinnin etuja

- Lähdekoodin virheettömyys, tiiviys ja selkeys
- Ylläpito- ja tuotetukikustannukset pienenevät
- Ohjelmointitaidot paranevat
- Eri ratkaisuvaihtoehtoja pohditaan perusteellisemmin
- Tieto ei ole vain yhden henkilön varassa
- Ohjelmoijien osaaminen leviää jopa koko tiimille, jos pareja vaihdetaan säännöllisesti
- Keskittyminen työntekoon on parempaa, kun toinen on mukana seuraamassa

Pariohjelmoinnin haasteita

- Tehokkuus
- Etätyöskentely
- Vaativuus
- Driverin ja copilotin ohjelmointitaitojen vastaavuus
- Copilotin rooli
- Henkilökemiat
- Osapuolten keskinäinen kommunikointi

Kysely: Pariohjelmoinnin aloittaminen

Valitse itsellesi sopivin vaihtoehto kustakin kysymyksestä. Kiitos vastauksistasi!

1. Olen *

- ☐ mies.
☐ nainen.

2. Olen *

- ☐ 19-21 vuotta.
☐ 22-24 vuotta.
☐ 25 vuotta tai vanhempi.

3. Opiskelen tietojenkäsittelyä *

- ☐ 1. vuotta.
☐ 2. vuotta.
☐ 3. vuotta.
☐ 4. vuotta tai kauemmin.

4. Ohjelmoinnin opiskelu on minusta yleensä *

- ☐ todella helppoa.
☐ melko helppoa.
☐ ei helppoa eikä vaikeaa.
☐ melko vaikeaa.
☐ todella vaikeaa.

5. Pariohjelmointi on minulle *

- ☐ täysin uusi menetelmä.
☐ hieman uusi menetelmä.
☐ hieman tuttu menetelmä.
☐ täysin tuttu menetelmä.

6. Ohjelmamateriaali ja opastus pariohjelmointiin oli mielestäni *

- ☐ selkeä ja riittävä.
☐ melko selkeä ja riittävä.
☐ hieman epäselvä ja riittämätön.
☐ epäselvä ja riittämätön.
☐ Muu:

7. Pariohjelmointi vaikuttaa minusta *

- ☐ toimivalta ja kokeilemisenarvoiselta menetelmältä.
☐ jokseenkin toimivalta ja kokeilemisenarvoiselta menetelmältä.
☐ jokseenkin toimimattomalta ja turhalta menetelmältä.
☐ toimimattomalta ja turhalta menetelmältä.
☐ Muu:

8. Parien muodostus opettajan johdolla alkukartoituksen pohjalta vaikuttaa *

- ☐ toimivalta ratkaisulta.
☐ jokseenkin toimivalta ratkaisulta.
☐ jokseenkin toimimattomalta ratkaisulta.
☐ toimimattomalta ratkaisulta.
☐ Muu:

9. Opintojakson suorittaminen pariohjelmoinnilla *

- ☐ todennäköisesti auttaa ja kannustaa minua suoriutumaan opintojakson vaatimuksista.
☐ ei todennäköisesti vaikuta suoriutumiseeni opintojakson vaatimuksista.
☐ todennäköisesti lannistaa ja vaikeuttaa suoriutumaan opintojakson vaatimuksista.
☐ Muu:

10. Haluaisin vielä sanoa, että...

Voit kirjoittaa tekstikenttään ajatuksiasi, näkemyksiäsi tai kysymyksiä pariohjelmointiin tai tähän kyselyyn liittyen.

Kysely: Pariohjelmoinnin toteuttaminen

Valitse itsellesi sopivin vaihtoehto kustakin kysymyksestä. Kiitos vastauksistasi!

1. Olen *

- ☐ mies.
☐ nainen.

2. Olen *

- ☐ 19-21 vuotta.
☐ 22-24 vuotta.
☐ 25 vuotta tai vanhempi.

3. Olo-ohjelmoinnin opintojakso on ollut tähän saakka *

- ☐ todella helppo.
☐ melko helppo.
☐ sopiva vaikeustasoltaan.
☐ melko haastava.
☐ todella haastava.

4. Suhtautumiseni pariohjelmointiin on *

- ☐ positiivisempi kuin opintojakson alussa.
☐ yhtä positiivinen kuin opintojakson alussa.
☐ yhtä neutraali kuin opintojakson alussa.
☐ yhtä negatiivinen kuin opintojakson alussa.
☐ negatiivisempi kuin opintojakson alussa.

5. Pariohjelmointi *

- ☐ on vähentänyt poissaolojani ja parantanut keskittymistäni oppitunneilla.
☐ ei ole vaikuttanut poissaoloihini tai keskittymiseeni oppitunneilla.
☐ on lisännyt poissaolojani ja heikentänyt keskittymistäni oppitunneilla.

6. Olen ohjelmoitunut parin kanssa *

- ☐ koko ajan.
☐ suurimman osan ajasta.
☐ vain opintojakson alussa.
☐ yhden tai kahden oppitunnin ajan.

7. Tärkein syy siihen, että olen jatkanut ohjelmoinnista parin kanssa on

Vastaa tähän kysymykseen, jos valitsit 6. kysymyksestä ensimmäisen tai toisen vaihtoehdon.

- ☐ parinmuodostuksen onnistuminen.
☐ positiivinen suhtautuminen parityöskentelyyn.
☐ henkilökkemioiden toimivuus.
☐ ohjelmointitaitojen parantuminen pariohjelmoinnilla.
☐ vaihtelu itsenäiselle työskentelylle.
☐ opintojakson haasteellisuus.

☐ Muu:

8. Tärkein syy siihen, että en ole jatkanut ohjelmoinnista parin kanssa on

Vastaa tähän kysymykseen, jos valitsit 6. kysymyksestä kolmannen tai neljännen vaihtoehdon.

- ☐ parinmuodostuksen epäonnistuminen.
☐ negatiivinen suhtautuminen parityöskentelyyn.
☐ henkilökkemioiden toimimattomuus.
☐ puutteet omissa ohjelmointitaidoissa.
☐ puutteet parin ohjelmointitaidoissa.
☐ opetustilan soveltumattomuus pariohjelmointiin.
☐ opintojakson haasteellisuus.

☐ Muu:

9. Opintojakson jatkimmalla puolella aion *

- ☐ jatkaa pariohjelmointia saman parin kanssa.
☐ jatkaa pariohjelmointia toisen parin kanssa.
☐ jatkaa ohjelmointia itsenäisesti.

☐ Muu:

10. Lisäksi haluaisin sanoa, että

Voit kirjoittaa tekstikenttään ajatuksiasi, näkemyksiäsi tai kysymyksiä pariohjelmointiin tai tähän kyselyyn liittyen.

Kysely: Pariohjelmoinnin päättäminen

Valitse itsellesi sopivin vaihtoehto monivalintakysymyksistä ja vastaa avokysymyksiin omien kokemustesi mukaan. Kiitos vastauksistasi!

1. Olen *

- ☐ mies.
☐ nainen.

2. Olen *

- ☐ 19-21 vuotta.
☐ 22-24 vuotta.
☐ 25 vuotta tai vanhempi.

3. Menestyin opintojakson ensimmäisessä tentissä *

- ☐ kiittävästi.
☐ hyvin.
☐ tyydyttävästi.
☐ välttävästi.
☐ heikosti.
☐ En osallistunut tenttiin.

4. Olen ohjelmoinut parin kanssa opintojakson jälkimmäisellä puoliskolla *

- ☐ koko ajan.
☐ suurimman osan ajasta.
☐ satunnaisesti.
☐ en lainkaan.

5. Pariohjelmointi *

- ☐ on auttanut ja kannustanut minua suoriutumaan opintojakson vaatimuksista.
☐ ei ole vaikuttanut suoriutumiseeni opintojakson vaatimuksista.
☐ on lannistanut ja vaikeuttanut suoriutumistani opintojakson vaatimuksista.
☐ Muu:

6. Tämän opintojakson pariohjelmointikokemusten perusteella *

- ☐ hyödynnän pariohjelmointia myös tulevilla ohjelmointikursseilla ja/tai työelämässä.
☐ harkitsen hyödyntäväni pariohjelmointia myös tulevilla ohjelmointikursseilla ja/tai työelämässä.
☐ en todennäköisesti hyödynnä pariohjelmointia tulevilla ohjelmointikursseilla ja/tai työelämässä.
☐ en hyödynnä pariohjelmointia tulevilla ohjelmointikursseilla ja/tai työelämässä.
☐ Muu:

7. Jos olet ohjelmoinut parin kanssa joko koko ajan tai suurimman osan ajasta tällä opintojaksolla, kerro tärkeimmät syyt pariohjelmoinnin hyödyntämiseen.

8. Jos et ole ohjelmoinut parin kanssa juuri lainkaan tai pariohjelmointi on ollut satunnaista tällä opintojaksolla, kerro tärkeimmät syyt itsenäiseen työskentelyyn.

9. Miten pariohjelmointia voisi mielestäsi vielä kehittää tällä opintojaksolla? *

10. Haluaisin vielä sanoa, että...

Voit kirjoittaa tekstikenttään ajatuksiasi, näkemyksiäsi ja kysymyksiä pariohjelmointiin tai tähän kyselyyn liittyen.

Kysely: Pariohjelmoinnin hyödyntäminen opintojakson päättymisen jälkeen

Valitse itsellesi sopivin vaihtoehto monivalintakysymyksistä ja vastaa avokysymyksiin omien kokemustesi mukaan. Kiitos vastauksistasi!

1. Olen *

- ☐ mies.
- ☐ nainen.

2. Olen *

- ☐ 19-21 vuotta.
- ☐ 22-24 vuotta.
- ☐ 25 vuotta tai vanhempi.

3. Sain olio-ohjelmoinnin opintojaksosta arvosanan *

- ☐ K5
- ☐ H4
- ☐ H3
- ☐ T2
- ☐ T1
- ☐ Hylätty

4. Valitse sopivin vaihtoehto *

- ☐ Olen hyödyntänyt pariohjelmointia olio-ohjelmoinnin opintojakson päättymisen jälkeen.
- ☐ En ole hyödyntänyt pariohjelmointia olio-ohjelmoinnin opintojakson päättymisen jälkeen.

5. Jos olet hyödyntänyt pariohjelmointia opintojakson päättymisen jälkeen, kerro miten ja miksi.

6. Kerro, mitä mieltä olet pariohjelmoinnista nyt. Onko suhtautumisesi pariohjelmointiin muuttunut alkusyksyyn 2012 verrattuna? *

7. Pitäisikö pariohjelmointia mielestäsi hyödyntää ohjelmoinnin opetuksessa laajemminkin? Miksi? Miksi ei? *

8. Haluaisin vielä sanoa, että...

Voit kirjoittaa tekstikenttään ajatuksiasi, näkemyksiäsi ja kysymyksiä pariohjelmointiin tai tähän kyselyyn liittyen.

Ensimmäinen kysely: kysymys 10

"Aika epäilevin mielin lähden tähän pariohjelmointikokeiluun. Jos parin olisi saanut päättää itse niin ei olisi mitään ongelmaa, mutta jos henkilökemiat eivät toimi niin koko kurssin suorittaminen voi olla varsin ankea kokonaisuus."

"Kaikenlaista paskaa sitä ihmisten kiusaksi keksitäänkään."

"Lähtötasotesteistä tulisi kertoa vaikka viikko ennen kuin niitä lähdetään tekemään jottei esimerkiksi allekirjoittanut poistasi kaikkia java-harjoituksia koneelta. Joillakin on huonommat lähtökohdat tehdä testejä materiaalisesti."

"Kuten sanoinikin jo, parien valinta järjestely on vähän pöljä. Parhaat ohjelmoijat ovat pari, ja surkeimmat ovat pari. Se on epäreilua niitä kohtaan jotka eivät osaa kunnolla. Heikommilla oppilailla pitäisi olla parina joku sellainen joka osaa edes jotain ja osaisi ohjata pariaan ja opettaa tätä."

"Olen erittäin skeptinen pariohjelmointia kohtaan opiskelumenetelmänä. Silloin kun oikeasti tekee jonkun kanssa sovellusta se todennäköisesti toimii hyvin, mutta nyt opetellaan kummallekin mahdollisesti uutta asiaa, jolloin pitäisi minusta ensin osata tehdä itse..."

"Pariohjelmoinnissa olennaista on että parin ohjelmointitaidot ovat jokseenkin samalla tasolla ja että henkilökemia toimii."

Toinen kysely: kysymys 10

"Olin aluksi aika skeptinen koko pariohjelmoinnin suhteen, mutta kun kohdalleni sattui tuttu kaveri joka on suurin piirtein samalla tasolla koodauksen kanssa niin pariohjelmointi on toiminut. Mutta jos pariksi olisi tullut joku hiljaisempi kaveri jonka kanssa ei pääse samalle aaltopituudelle, olisin varmaan tehnyt hommia enemmän yksikseen."

"Pariohjelmointi 5/5."

”Pariohjelmointi saattaa olla tehokas keino ja ymmärrän sen hyödyt, mutta kun opetellaan uutta vaikeaa asiaa, niin itselleni ainakin on tärkeää saada tehdä itse eikä katsoa vieressä.”

”Hyvä, ettei kuitenkaan jouduttu menemään käsikynkkää vessaan.”

”Kurssi on ollut tuntien peruuntumisten takia katkonainen ja parityöskentelyä on tämän takia ollut vähemmän. Tästä syystä pariohjelmointia ei ole ehditty paljoa harrastaa. Parityöskentelyn onnistuminen on kiinni suurelta osin henkilökemiasta ja parien lähtötasosta, eikä se välttämättä toimi jos parien osaaminen ei ole suurin piirtein samalla tasolla.

”Kurssi on ollut haastava, ja on ollut paljon uutta asiaa, jonka ymmärtämiseksi on ollut pakko tehdä itsekkin töitä silloin kun olisi pitänyt olla co-pilot.”

Kolmas kysely: kysymys 7

”Pariohjelmointia suositeltiin opintojakson alussa. Pariohjelmoinnin hyödyntäminen vähensi virheiden määrää ja nopeutti niiden löytämistä ohjelmakoodista. Asioista joista oli hieman epävarma, saattoi kysyä mielipidettä ohjelmointiparilta.”

”Mielenkiinto erilaista ohjelmointitapaa kohtaan. Huomioitavaa olisi että parit olisivat kutakuinkin samalla tasolla.”

”Parina oli vanha tuttu, jonka kanssa ajatukset olivat samalla taajuudella ja muutenkin taitotaso oli about sama. Tekeminen oli järkevää ja fiksua.”

”Hyvä pari”

”Parin kanssa on hyvä miettiä miten pääsisi eteenpäin, jumitilanteen tullessa. Myös virheet on helpompi huomata yhdessä kuin yksin.”

Kolmas kysely: kysymys 8

”Opintojakson ensimmäisellä puoliskolla en ohjelmoinut parin kanssa suuren tasoeron ja huonohkojen henkilökemioiden takia.”

”Uusintaan menee”

”Ensimmäinen pari oli pois puolet tunneista. Toisen parin kanssa päätettiin, että työskentelemme paremmin itsenäisesti, mutta kommunikoimme ja autamme toisiamme omien päätteidemme ääritä.”

”töissäkin pitää osata itte jotaki”

”Alussa parini ohjelmointitaidot eivät yltäneet lähellekkään omiani, joka osittain hidasti työskenteilyä varsinkin kun oli hänen vuoro ohjelmoida. Se ei kuitenkaan vaikuttanut suoriutumiseeni opintojaksolla.”

Kolmas kysely: kysymys 9

”Tasaisemmat parit ja mahdollisesti paritentejä”

”Tuntuu että kurssin alussa monellakaan ei parien muodostus osunut kohdalleen, mutta toisella yrittämällä onnistui paremmin.”

”Lopettamalla moisen pelleilyn”

”Vaikka pariohjelmoinnista on ollut hyötyä niin näen silti melko turhana arpoa pareja ja sanoa että ”te kaksi ohjelmoitte nyt kahdestaan”. Kavereilta tulee kysyttyä apua muutenkin, niin käytännössä pariohjelmointi on poikennut normaalista ohjelmoinnin tunnista vain siten että nyt on ohjelmoinut vain yhden ihmisen kanssa sen sijaan että useampi lähistöllä istuva alkaisi porukalla etsiä virhettä yhden ihmisen koodista.”

”Pareja voisi vaihtaa useammin”

”Jokin isompi esim. harjoitustyö-tyyppinen ohjelmointiprojekti olisi mielenkiintoinen pariohjelmointityö ja se saattaisi vaikuttaa positiivisesti ohjelmointiparin työskentelyyn ja yhteistyökykyyn.”

”Suosittelen pariohjelmoinnista luopumista. Se on teennäistä, humanisti-ituhippi-ideologien fantasiaa.”

”oppilaat voisi olla itsekin miettimässä omia parejaan”

”Mielestäni tasokokeista pitäisi ilmoittaa selvästi etukäteen opiskelijoille, joiden perusteella parit muodostetaan. Joillakin voi olla selvästi huonommat lähtökohdat tasokokeisiin, jos esimerkiksi vanhat harjoitukset on poistettu koneelta ja yksinkertaisimmatkin asiat on muistuteltava mieleen googlettamalla. Näinhän tällä kurssilla sattui, kun javaa olin viimeksi ohjelmoinut noin 8 kuukautta aikaisemmin ja vanhoja harjoituksia ei enää ollut. Jos olisin tiennyt tasokokeesta ja pariohjelmoinnista, olisin nostanut java-osaamiseni vanhalle tasolle.”

Kolmas kysely: kysymys 10

”Pariohjelmointi toimii mielestäni ehkä parhaiten työelämässä, ja vain osittain opiskelijoille. Aina-kin itse olen sellainen tyyppi että eivät opetukset jää niin hyvin mieleen jos en ole itse konkreettisesti sitä tekemässä ja ITSE ratkaissut sitä mielessäni pätkäilyn jälkeen. Työelämässä tietenkin myös opitaan uutta, mutta vähemmässä määrin ja tekeminen on erilaista”

”Pariohjelmointi on toimiva menetelmä ja toiminut muilla kursseilla, mutta koska tämä ko kurssi oli itselle ja parilleni haastava se ei toiminut. Kumpikaan ei pysty neuvomaan toista”

Neljäs kysely: kysymys 5

”Hyvä pari kurssilla, jonka kanssa ollut samassa projektissa. Pari ohjelmoinnin hyödyllisyys riippuu mielestäni parien tasosta”

”melkein kaikessa ohjelmoinnissa tullut hyödynnettyä”

”Hyödynsimme pariohjelmointia työelämälähtöisessä projektissa. Meitä oli isompi ryhmä ja jakaudimme pareiksi tekemään töitä.”

Neljäs kysely: kysymys 6

”Huomattavasti parempi mielikuva.”

”on muuttunut. Huomasin kuinka toimivaa ja tehokastaa se on. Se vähentää virheitä koodissa huomattavasti”

"Pariohjelmointi on hyödyllistä sopivan parin kanssa, kun kumpikin voi oppia toisiltaan ja auttaa tarvittaessa. Suhtautumiseni on muuttunut positiivisemmaksi."

"Ei"

"Ei toiminut ainakaan omalla osallani silloin, mutta miksei voisi toimia tulevaisuudessa työelämässä, kun ohjelmoijapari todennäköisemmin on ainakin suurimmaksi osaksi paikalla."

"Pariohjelmointi on ihan toimiva ja hyödyllinen menetelmä sillä usein kaveri huomaa virheet, joita ei itse muuten huomaisi. Tunneilla pariohjelmointia kuitenkin olisi pitänyt hyödyntää siten, että toinen koodaa ja toinen tekee ehdotuksia logiikan suhteen yms. Näen toimivampana ratkaisuna, että molemmat koodaavat ja vaikean kohdan tullessa esille, mietitään yhdessä miten sen saisi helpoimmin toteutettua. Sillä tavalla hyödynsin pariohjelmointia myös edellisessä kohdassa mainitussa projektissa."

"Aluksi en tiennyt mitä ajatella pariohjelmoinnista, mutta opintojakson aikana huomasin että se on hyödyllinen työskentely tapa kuhan parit on samalla viivalla ohjelmoinnissa."

Neljäs kysely: kysymys 7

"Kyllä, muttei aluksi vaan vasta kun kaikilla on taitoa jota hyödyntää ja jotta parit voidaan yhdistellä aiemman taidon näytteen perusteella"

"pitäisi. Saadaan nopeampaa toimivampaa koodia.. debuggauksen tarve vähenee"

"Pariohjelmointia voisi olla jossain määrin ohjelmointiopinnoissa. Ei välttämättä koko aikaa, mutta välillä."

"Mielestäni pariohjelmoinnissa on etuja yksin koodaamiseen. Sitä voitaisiin mielestäni käyttää ohjelmoinnin opetuksessa alusta pitäen JOS molemmat osapuolet haluavat työskennellä parina."

"Ei. Lähtökodot ohjelmointiin opiskelijoilla niin erit, että pariohjelmointi harvalla parilla oikeasti toimisi."

”Javan perusteet (ensimmäinen ohjelmoinnin kurssi joka meillä koulussa oli) olisi hyvä hoitaa yksilötyönä, mutta siitä eteenpäin kunhan ohjelmointi alkaa olla tuttua, pariohjelmointi voi olla ihan toimiva ratkaisu, jos vain molemmat saavat koodata sen sijaan, että toisen pitäisi vain istua vieressä tuijottelemassa ja huomauttaa jostain kirjoitusvirheistä jos parilla sujuu hyvin ja nopeasti.”

”Voi sitä hyödyntää kunhan löytyy parit jotka on saman tasoisia ja tulee toimeen toistensa kanssa, mutta pakottamalla pariohjelmointia ei pidä tehdä”

1. Kuinka paljon olit perehtynyt pariohjelmointiin työskentelymuotona ennen suostumistasi pariohjelmointikokeiluun olio-ohjelmoinnin opintojaksolla?
2. Oliko ennen opintojakson alkua saamasi perehdytysmateriaali mielestäsi selkeä ja riittävä? Miten materiaalia voisi vielä kehittää?
3. Kuinka hyvin opetustila sopi mielestäsi pariohjelmointiin? Millainen olisi paras mahdollinen opetustila pariohjelmointia ajatellen?
4. Onnistuiko opiskelijoiden jakaminen tasavahvoihin pareihin alkukartoituksen pohjalta? Millaiset parit toimivat näkemyksesi mukaan parhaiten?
5. Miten opiskelijat suhtautuivat pariohjelmointiin? Muuttuivatko käsitykset opintojakson kuluessa?
6. Kuinka hyvin opiskelijat noudattivat pariohjelmoinnin periaatteita? Kommunikoivatko parit riittävästi keskenään?
7. Vaikuttiko pariohjelmointi poissaolojen määrään tai keskittymiseen oppitunneilla?
8. Millaisille opiskelijoille pariohjelmointi sopii työskentelymuotona parhaiten tämän kokeilun perusteella?
9. Miten opiskelijat menestyivät verrattuna aikaisempiin toteutuskertoihin? Uskotko pariohjelmoinnilla olleen vaikutusta menestykseen?
10. Millainen oli läpäisyprosentti verrattuna aikaisempiin toteutuskertoihin? Uskotko pariohjelmoinnilla olleen vaikutusta läpäisyprosenttiin?
11. Oliko arvosanoissa havaittavissa eroja pareittain ja yksin ohjelmoineiden välillä? Uskotko työskentelymuodolla olleen vaikutusta arvosanoihin?
12. Miten pariohjelmointia voisi vielä mielestäsi kehittää kyseisellä opintojaksolla?
13. Erosiko opettajan rooli pariohjelmointia käytettäessä aikaisempaan opetustapaan verrattuna? Helpotti-ko pariohjelmointi opettajan työtä vai aiheutuiko siitä lisävaivaa?
14. Voisitko käyttää pariohjelmointia opetusmenetelmänä myös jatkossa? Miksi? /Miksi et?
15. Haluaisitko vielä lisätä jotain?